Yuachun Wang

State of California
The Resources Agency
Department of Water Resources
Office of State Water Project Planning

METHODOLOGY FOR FLOW AND SALINITY ESTIMATES IN THE SACRAMENTO-SAN JOAQUIN DELTA AND SUISUN MARSH

TWENTY-FIRST ANNUAL PROGRESS REPORT TO THE STATE WATER RESOURCES CONTROL BOARD
In Accordance with Water Right Decision 1485, Order 9

June 2000

Gray Davis	**Mary D. Nichols**	**Thomas M. Hannigan**
Governor	Secretary for Resources	Director
State of California	The Resources Agency	Department of Water Resources

D—0 3 8 5 0 3

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

FOREWORD

This is the twenty-first annual progress report of the California Department of Water Resources' San Francisco Bay-Delta Evaluation Program, which is carried out by the Delta Modeling Section. It documents progress in the development and enhancement of the Delta Modeling Section's computer models and reports the latest findings of studies conducted as part of the program. This report was compiled by Michael Mierzwa under the direction of Paul Hutton, program manager for the Bay-Delta Evaluation Program.

For more information contact:

Paul Hutton
hutton@water.ca.gov
(916) 653-5601

or

Michael Mierzwa
mmierzwa@water.ca.gov
(916) 653-9794

1 Introduction

For the last seven years, the Delta Modeling Section has been developing and enhancing its Delta Simulation Model (DSM2) and support tools. The following are brief summaries of work that has been conducted during the past year. The names of the contributing authors are in parentheses.

Chapter 2 – DSM2 Schematic Viewing Program

This chapter introduces the DSM2 Schematic Viewing Program (DSVP), a new graphical user interface that may be used to view model input and output data. DSVP allows the user to view time series data through an animation routine. Section staff is using a beta version of DSVP; however, further development is ongoing. (*Tawnly Pranger*)

Chapter 3 – DSM2 San Joaquin River Boundary Extension

This chapter describes the Section's initial efforts to extend the DSM2 San Joaquin River (SJR) boundary upstream of Vernalis. Many Delta water supply, water quality, and fishery issues are closely linked to conditions in the San Joaquin River. Extension of the boundary will provide a tool to investigate how the Delta may respond to different SJR management strategies. Channel geometry has been developed along the SJR from Vernalis upstream to the Bear Creek confluence near Stevinson. Several trial runs of DSM2-HYDRO have been conducted to test and refine the channel geometry. In coordination with the San Joaquin River Management Program's Water Quality Subcommittee, historical data are being collected to calibrate the boundary extension. (*Thomas Pate*)

Chapter 4 – VPlotter

This chapter describes VPlotter, the latest in the group of VISTA tools for retrieving, manipulating, and managing time series data. VPlotter builds upon existing functionality of VISTA and VScript. VPlotter focuses on repeatability and automation. (*Nicky Sandhu*)

Chapter 5 – DSM2 Particle Tracking Model Development

This chapter provides an update on DSM2-PTM development efforts. PTM simulations have primarily been limited to neutrally buoyant particles. Work during the past year has focused on incorporating more sophisticated particle behavior characteristics and developing a graphical user interface to define and modify these characteristics. The following behaviors have been incorporated: life stage or phase; development time; fall velocity; mortality rate; and vertical positioning. Behavior characteristics that are in the process of being implemented include transverse positioning, flow positioning, and water quality responses. (*Aaron Miller*)

Chapter 6 – Effects of Salinity-Induced Density Variation on DSM2 Hydrodynamics

This chapter discusses the results of a DSM2 experiment conducted to measure the effects of a salinity-induced feedback on hydrodynamic results. Results suggest that for typical model applications, baroclinic effects may be ignored as they are mild compared with model error. Baroclinic effects depend strongly on hydrology but weakly on location. (*Eli Ateljevich*)

Chapter 7 – Artificial Neural Network Development

This chapter reviews progress made in development and use of Artificial Neural Networks for estimating salinity at various locations in the Delta. An ANN was developed to estimate DSM2 tidal boundary salinity at Martinez for forecasting applications. (*Tawnly Pranger*)

Chapter 8 – Filling In and Forecasting DSM2 Tidal Boundary Stage

This chapter describes a new approach to fill in missing historical and forecast future Martinez stage values. An adequate characterization of stage at Martinez, the DSM2 downstream tidal boundary, is critical in order for HYDRO to simulate Delta hydrodynamics accurately. The approach combines a traditional astronomical tide model that predicts periodic fluctuations with a vector autoregressive model that predicts non-periodic residue. The approach produces fill-in values that are extremely accurate, regardless of events which take place during the fill-in period. The approach also improves tide forecasts, with the largest improvements during the first week. (*Eli Ateljevich*)

Chapter 9 – Dissolved Oxygen Modeling Using DSM2-QUAL

This chapter summarizes work conducted in preparation of a dissolved oxygen calibration in the vicinity of Stockton on the San Joaquin River. DSM2-QUAL was updated to reflect changes in hydrodynamics and general input/output modules. Calibration efforts will be conducted in coordination with the San Joaquin River Total Maximum Daily Load (TMDL) stakeholder process. (*Hari Rajbhandari*)

Chapter 10 – DSM2 Recalibration

This chapter describes the Section's participation in an Interagency Ecological Program project work team to recalibrate DSM2-HYDRO and QUAL with new geometry and flow data. The team, including staff from DWR (Department of Water Resources), USGS (U.S. Geological Survey), USBR (U.S. Bureau of Reclamation), CCWD (Contra Costa Water District), MWDSC (Metropolitan Water District of Southern California), and Stanford University, was formed in late 1998 and posted initial results in 1999. The team plans to have both HYDRO and QUAL calibrated by middle of summer 2000. Intermediate results may be viewed on a publicly available website.
(*Parviz Nader-Tehrani and Bijaya Shrestha*)

Chapter 11 – DSM2-QUAL Initialization

This chapter surveys methods of initializing DSM2-QUAL real-time model runs. When the model simulation period is long compared to system memory, initial conditions influence only a small fraction of the run. In such cases (e.g. planning studies), the model is typically "cold started" from a numerically convenient initial condition. This approach, which relies on long-term system memory loss, is not sufficiently accurate for real-time model applications. Two "warm start" schemes, an "optimization-based" approach and a "patch-based snapshot" approach, were developed to generate initial conditions for real-time applications. Both are shown to provide reasonably good initial conditions; both schemes are shown to be superior to the "cold start" approach.
(*Eli Ateljevich*)

2 DSM2 Schematic Viewing Program

2.1 Introduction

This chapter introduces the DSM2 Schematic Viewing Program (DSVP). This program is designed to serve as a graphical pre/post processor for DSM2 and its associated files. The DSVP is written in Java, an object-oriented, platform-independent language. For the purposes of this chapter, any words included between <> indicate the name of a button on the interface.

2.2 DSVP Input Files

2.2.1 DSM2 Input Files

The input files for DSM2 are in an ASCII column based structure. The current version of the DSVP uses the channels.inp file to gather information about the schematic channels. The program can also use the translations.inp file to assign data to a particular channel from a specified HEC-DSS file pathname. Future versions will use the data from all of the input files and allow manipulation of those files.

2.2.2 DSM2 Output

Both DSM2-HYDRO and QUAL output their results to either the HEC-DSS file format or as an unformatted binary tidefile. The current version of DSVP can read and display time series data from the HEC-DSS format. Future versions of the program will be able to read and display data from binary tidefiles as well.

2.2.3 Coordinate Data: .xy

The only data file that is required by the DSVP that is not an associated DSM2 file is a list of UTM coordinates for each node. This file is used to place each node in relation to the other nodes on the schematic.

2.3 Using the DSVP

2.3.1 General Description

The DSVP interface (Figure 2-1) consists of a control panel on the left side and a schematic display on the right.

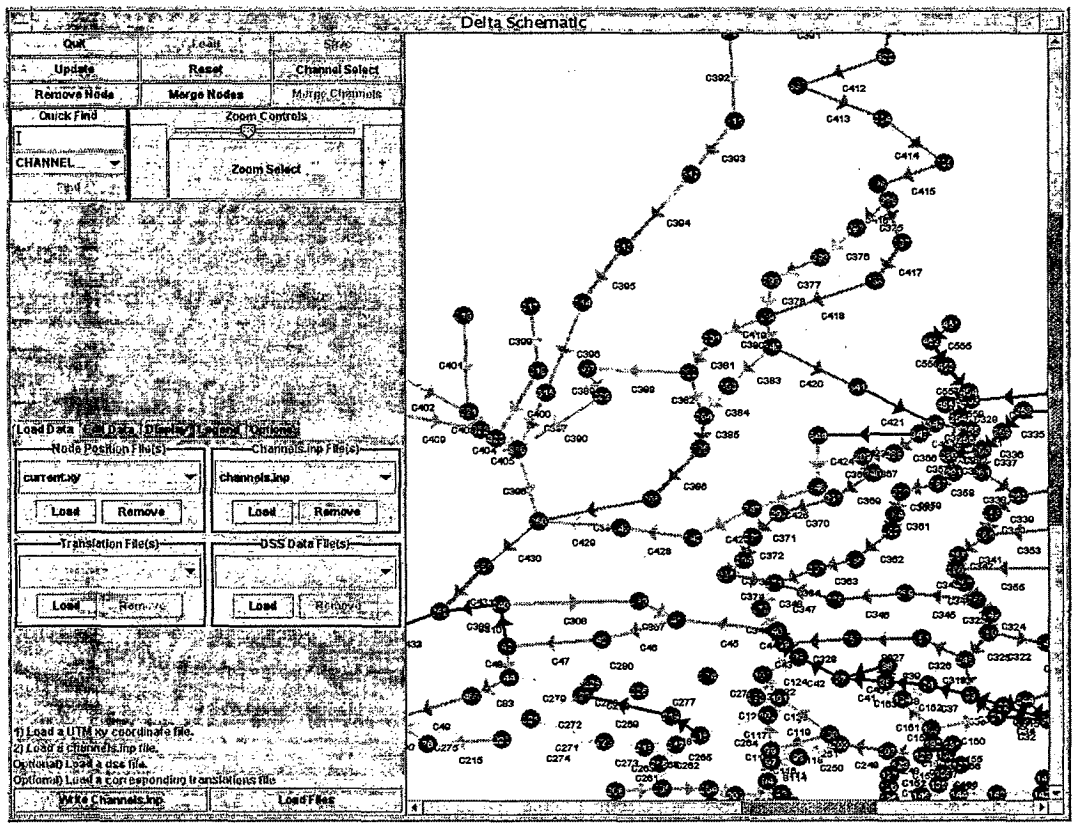

Figure 2-1: DSVP Interface

The top of the control panel houses nine buttons with various functions (from left to right and top to bottom: Quit, Load, Save, Update, Reset, Channel Select, Remove Node, Merge Nodes, Merge Channels). The Load, Save, and Merge Channels buttons have no functional purpose at this time and serve only as placeholders. Below the buttons are two panels, one is the Quick Find panel and the other is the Zoom Control.

The bottom of the control panel houses a tabbed pane with five panels. These panels are: Load Data, Edit Data, Display, Legend, and Options.

2.3.1.1 The Load Data Panel

The Load Data Panel (Figure 2-2) consists of four panels to load each of four different types of files. These files are: the UTM xy position file, channels.inp file, DSS file, and translation file. The UTM xy file and channels.inp file are required to run the DSVP. The translations.inp file may be required only if a DSS file requires it. A DSS file is only required to view time series data. More than one file may be loaded for each type. The UTM, channels.inp and DSS files will overwrite existing values from previously loaded files. The translations file will append values. Each file loading panel contains two buttons, one for loading files, the other for removing files. The Load Data Panel also contains two buttons. The first is the <Load Files> button, which loads the data from the selected files. The second is the <Write Channels.inp> button, which will write out a specified channels.inp file for the loaded and updated channels.

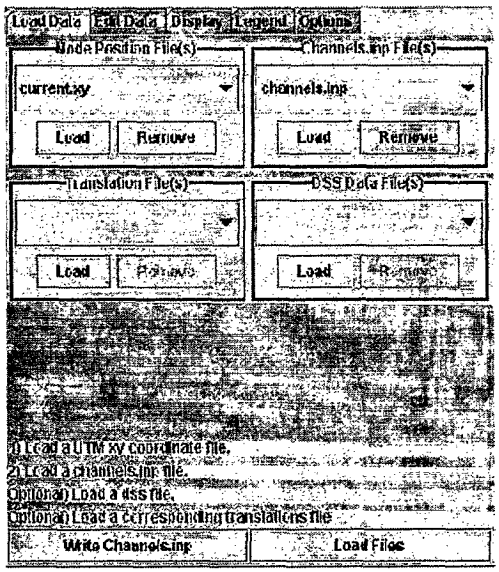

Figure 2-2: Load Data Panel

2.3.1.2 The Edit Data Panel

The Edit Data Panel (Figure 2-3) contains the polygon selection tool. This tool allows the user to create a polygon and retrieve the channels and nodes contained within the polygon. It can then list those channels and/or nodes as well as highlight the selected channels. It allows for the selection and deletion of channels from the list. It allows for the user to edit data and visual properties for a large number of channels.

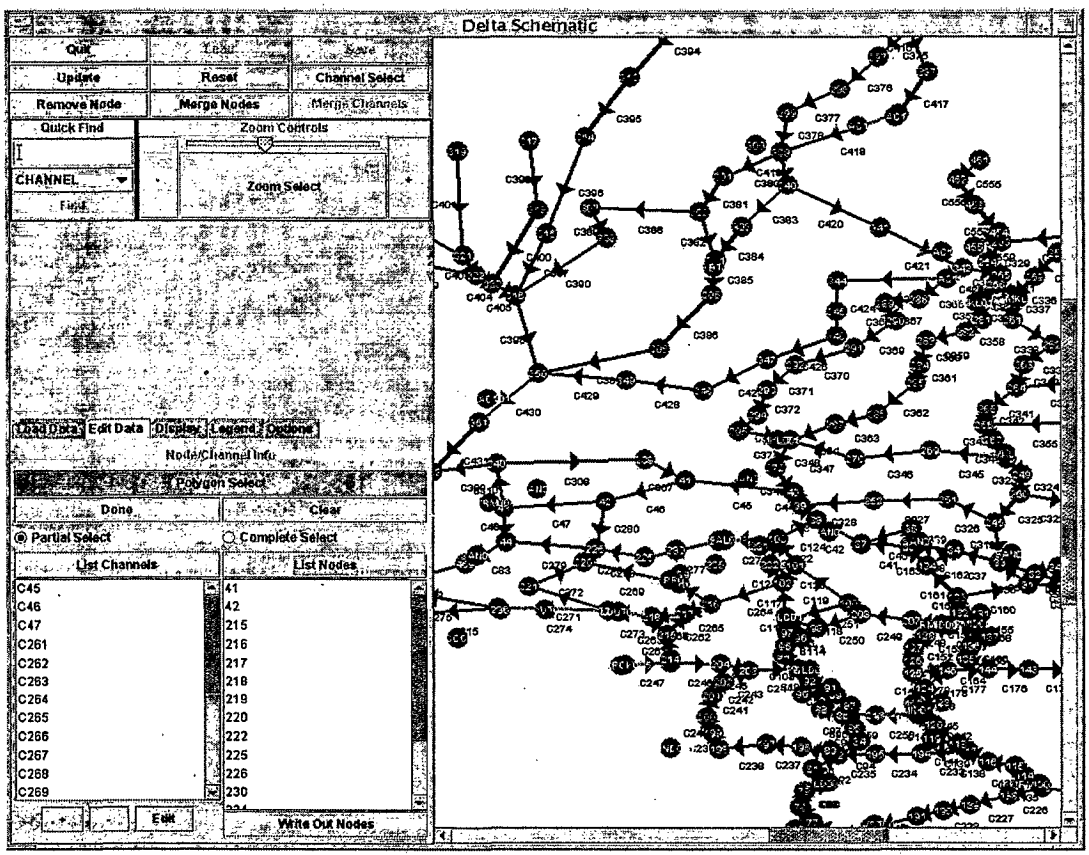

Figure 2-3: Edit Data Panel

2.3.1.3 The Display Panel

The Display Panel (Figure 2-4) contains three combo boxes that allow for the choice of available data types. The value displayed by the channel color, channel arrow length and channel bar is determined from the combo boxes. The checkboxes adjacent to the combo box allow the user to decide whether or not to display values by the associated method.

The Display Panel has four buttons that can either highlight channels that have translations or DSS data loaded or create a list of those channels depending on which button is selected.

The bottom of the Display Panel contains a panel for the loading of selected data types. If time series data have been loaded, the time at which to start displaying the time series can be selected (through use of the slider). Starting, pausing, and resetting the display of the time series can be done by selecting the appropriate button. This panel displays the current time and number of DSS paths containing data at the current time being displayed. It displays the maximum time and minimum time available from the DSS file (if any are loaded). If there are multiple time intervals, the interval to be displayed can be changed through the combo box.

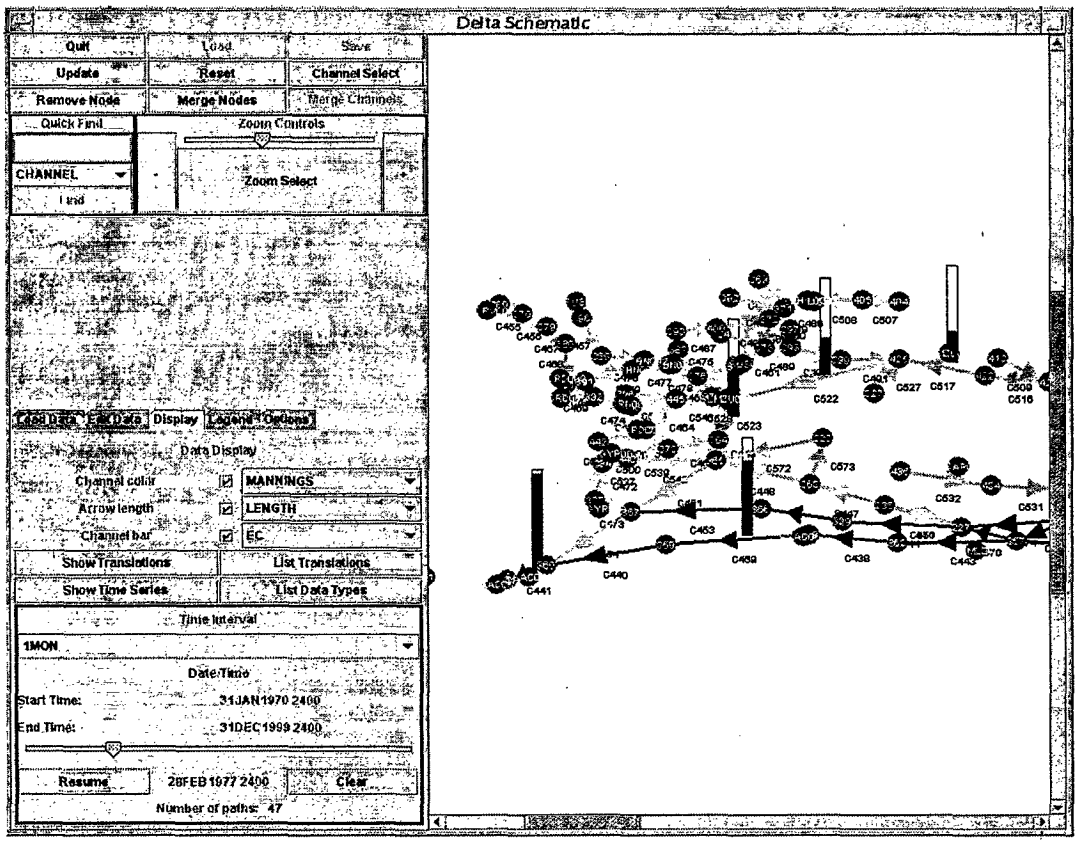

Figure 2-4: Display Panel

2.3.1.4 The Legend Panel

The Legend Panel (Figure 2-5) consists of a tabbed pane that has three panels (one for each data display type). The legend shows the range of values for the display type.

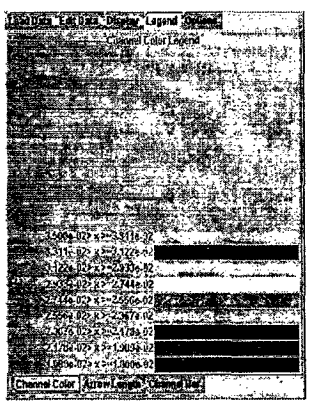

Figure 2-5: Legend Panel

2.3.1.5 The Options Panel

The Options Panel (Figure 2-6) consists of five toggle buttons that turn display features on or off. These are: node labels, channel labels, channel arrows, channel bars and null values. Null values occur when there is no valid data present for a channel. A sixth button updates the display after options have been changed. Changes will not appear until the <Apply Changes> button is used or the schematic is redrawn by some other request.

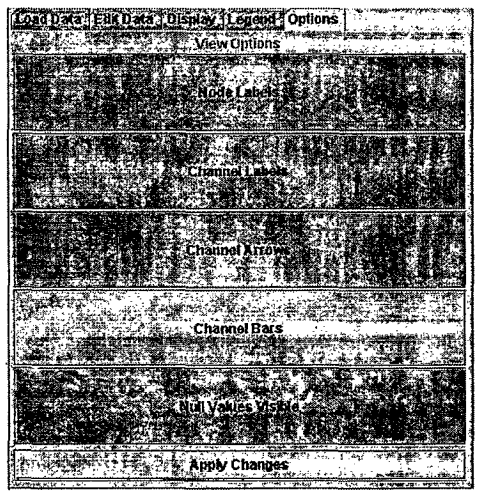

Figure 2-6: Options Panel

2.3.2 Loading Files

To use DSVP, a UTM xy file and a channels.inp file must be loaded. A DSS file needs to be loaded if time series data are to be displayed. A translations.inp file needs to be loaded if required for a DSS file. Translations are treated as aliases for existing channels and cannot exist on their own.

A file can be loaded by choosing an appropriate file type panel and clicking on its <Load> button. This launches a dialog that allows the user to select a file. The dialog will load with a default filter appropriate for the file type, but "all files (*.*)" can be selected if the file desired does not match the default filter.

If a DSS file is being loaded, once the file has been selected, a second frame will open with a list of the available paths for that file. This frame is a standard VISTA group table and allows for the displaying of data for each path in either graphical plot or tabular format. It also allows for filtering of paths by path part or time window. Once all desired filtering has been done, the user selects which of the remaining paths are to be loaded and clicks on the <Load Paths> button at the bottom. If no paths are selected, all of the listed paths will be loaded.

Multiple files can be loaded for each file type. Each subsequently loaded file will appear in the view window of the combo box. Previously loaded files can be selected from the combo box. Files will be loaded in the order in which they are selected. Since conflicting values are overwritten, the last file selected will have all of its values present, but files selected earlier may have values overwritten.

To remove a file, select the file in the combo box so that it appears in the view window and then click the <Remove> button for that panel. Once all desired files have been selected, click on the <Load Files> button to load the data and create the schematic.

Once the necessary files have been loaded, the rest of the functions can be performed in any order.

2.3.3 Updating the Display

Anytime a change is made that affects the appearance of the schematic (display options, changing channel name or connecting nodes, etc.), the schematic display must be updated to show those changes on the screen. The program does not do this automatically (except when displaying data) to allow for multiple changes to be made by the user without taxing the system. As a result, when changes are made, the user must click on the <Update> button at the top right corner of the control panel, or on the <Update Display> button in the appropriate panel being accessed.

2.3.4 Channel Select

The Channel Select toggle button is located at the top of the control panel in the lower right corner of the series of six buttons. While the button is depressed, the user may click on a channel in the schematic and a Channel Info frame (Figure 2-7) for that channel will be displayed. Selecting is most effective near the middle of the channel where it will not be confused with node selection.

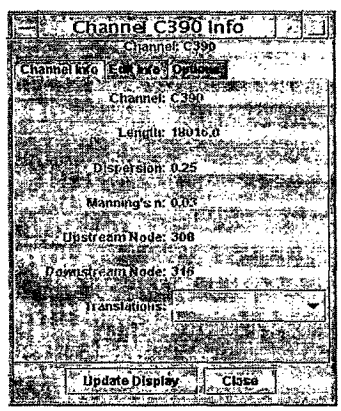

Figure 2-7: Channel Info Frame

The Channel Info frame consists of a tabbed pane with three panels. These panels are: Channel Info, Edit Info and Options.

The Channel Info panel allows the user to see the values that are held by the channel. These include the channel's name, length, dispersion, Manning's n, connecting nodes and any translations that are assigned to the channel.

The Edit Info panel allows the user to change the values contained by the channel. The values displayed in the Channel Info panel can be changed in this panel by overwriting the value in the text field and then clicking on <Apply Changes> to commit the changes to the channel. Translations can be added by entering a new name in the text field and clicking on <Add>. Translations can be removed by selecting the translation to be removed in the combo box and clicking on <Remove>. Be sure to click on <Apply Changes> when done.

The Options panel allows the user to change the display properties of an individual channel. The user can select whether or not to display data by channel color, channel arrow length, and/or channel bar. In addition, if the channel contains time series data, the user can select a type of data from the combo box and then graph that data by clicking the <Graph Data> button. The graph that is created is a direct extension of the VISTA graph.

The Update Display button can be used to apply the changes made on the visual appearance of the schematic.

The Channel Select works well in conjunction with either the Display Translations/Display Time Series buttons on the Display panel or with the Quick Find.

2.3.5 Quick Find

The Quick Find panel is found at the upper left side of the control panel. The Quick Find allows the user to quickly find the location of a channel or node. A channel may be located either by its name or by its translation. The method of searching (CHANNEL name, NODE name, or channel TRANSLATION) is selected from the combo box. The name to search for is entered into the text field. The user can click on <Find> and the schematic will be centered upon the location of the found object and the object will be highlighted red. The value entered must be an exact match for an existing name or no action will occur.

2.3.6 Zooming

The Zoom Controls panel is found at the upper right side of the control panel. The Zoom Controls panel allows for zooming into and out of the schematic by increment through the <+> and <-> buttons. It also allows for an absolute zooming between the increment range through the slider.

Zooming into a particular area can be accomplished by using the Zoom Select button. Once the button has been depressed, the user can click the mouse inside of the schematic to create the first corner of the zoom selection. With the mouse button still depressed, the user can drag the mouse to create the second corner of the selection rectangle. When the mouse button is released, the schematic will zoom into the selected area.

2.3.7 Polygon Selection and Editing

The Polygon Select tool is located in the Edit Data panel. When the <Polygon Select> button is depressed, the user can use a mouse to create a polygon in the schematic. Each time the mouse is clicked, a dot will appear at that point. If more than one dot is present, a line will appear between the newest dot and the one previously created. When all of the points have been created, click on <Done> to close the polygon. Once the polygon is closed, a list can be created. To create a new polygon, the user must first click on <Clear> to remove the previous points.

To create a list of nodes, simply click on <List Nodes> and an ordered list of all of the nodes contained within the polygon will be created and displayed below the button.

To create a list of channels, click on <List Channels> and an ordered list of the channels contained within the polygon will be created and displayed below the button. By selecting the <Partial Select> radio button option, the list of channels will contain all channels that have one or more nodes inside of the polygon. By selecting the <Complete Select> option, only those channels with both connecting nodes contained within the polygon will be selected. When the channels are selected, the channels in the list will be highlighted red in the schematic.

Once a list of channels has been created, additional channels can be added one at a time by using the <+> button. After this button is clicked, the first channel in the schematic that is clicked on is added to the list. To remove channels from the list, a channel or collection of channels is selected within the displayed list and then click on <->. By selecting one or more channels within the displayed list and clicking on <Edit> a new frame will open allowing for editing the values of the channels. If no channels are selected, <Edit> will open all channels within the list. This new frame allows the user to change the Manning's n and dispersion values as well as the display properties for the selected channels. Make sure to click <Apply Changes> if any changes are made to the values. There is currently no equivalent functionality for nodes.

2.3.8 Loading and Selecting Data Types

There are three ways to display values on the schematic. The first is by channel color, the second is by the length of the channel arrow, and the third is by the level of a bar graph located next to the channel. Any combination of the three display types can be

turned on or off by selecting or deselecting the appropriate checkbox next to the name. The data types to be displayed are chosen from the combo box next to each name. The list of data types is generated from any DSS files that are loaded. In addition, the static values from the channels.inp file are also loaded (dispersion, Manning's n and length).

Once the data types are selected and decisions on whether or not to use the display types are made, the data can be loaded by clicking <Load>. All static values will be loaded and displayed while time series data will be loaded into a buffer and the Load button will change to the Start button to indicate that the time series data are available for viewing.

2.3.9 Viewing Time Series Data

Once the data have been loaded by clicking on <Load> in the Display panel, the time series data can be viewed by clicking on <Start>. This starts the animation. The animation can be controlled by <Pause> and <Resume> (the Start button will change to these as appropriate). The absolute position in time can be selected through the slider. By clicking on the slider and dragging it to a position, the user can change the animation time period. While the slider is selected the number of paths containing time series data can be viewed at the bottom of the panel. Depending on the amount of data contained in the files, it may take a few seconds for the slider to update the values. When the user has decided on a time to continue with, the slider can be released and a new buffer will be created. The slider can also be used prior to starting the animation.

To load a new series of data types, first use <Clear> to reset the animation values. Then just repeat the earlier steps.

2.3.10 The Legend

The Legend panel simply shows what the data values are that correspond to a specific display type. The Legend panel consists of a tabbed pane containing three panels, one for each display type. By selecting a panel, the user can see the range of values attributed to that display type.

2.3.11 Changing Display Options

The Options panel allows the user to change display options for the schematic. There are five different choices that are selected by a corresponding toggle button. These choices are to display or not display the nodes, channel labels, channel arrows, channel bars and null values. The <Update Display> button can be clicked to update the visual appearance of the schematic after changes have been made.

When a channel value displayed by color has no value, it is displayed as black. By turning off the <Null Values> changes the color of those channels to the null color (light gray).

2.4 Current Directions

The DSVP is being rewritten to take advantage of the Java2D API. Java2D API is a more flexible architecture than the standard Java graphics package and is used for defining shapes, lines, and fonts. This allows for a wider range of graphics options. The DSVP has been split into modules to allow for development of functionality beyond the program itself and to ease future changes. Taking advantage of the reusable nature of object-oriented languages, the three main modules have split off from the DSVP. They are the Schematic, the DSM2 File Parser and the Tidefile Reader.

The Schematic is being initiated as a generic layout of nodes and arcs. This will allow for other programs such as CALSIM and DSM2-PTM to take advantage of the common graphic engine.

The DSM2 File Parser is designed to retrieve selected data from DSM2 input files and allow for easy graphical manipulation of the data contained in those files. The functions will include adding data, removing data, merging data and changing existing data. Additionally, the parser will allow checking the files.

The Tidefile Reader is being designed to function as a stand-alone application that will provide linkages for programs such as the DSVP or DSM2-PTM. The Tidefile Reader will use an XML (eXtensible Markup Language) based template structure (XTT - XML Tidefile Template) that will define the structure of the tidefile. The reader will also provide file exporting to either formatted ASCII or DSS format. Additionally, the reader will provide an animator to graphically view data over time.

The general layout of DSVP is being moved to a more menu driven layout. This allows for more space to be available for the schematic as well as increasing the logical structure of commands and operations.

In general, data is being moved to an XML structure. This format is flexible, allows for easy manipulation, is easy to discern its content and meaning outside of the program and has mechanisms for the self-checking of valid data.

3 DSM2 San Joaquin River Boundary Extension

3.1 Introduction

The purpose of the DSM2 model boundary extension is to create a direct dynamic link between the Delta and the State's second longest river, the San Joaquin. Many Delta water supply, water quality, and fishery issues are closely linked to conditions in the San Joaquin River (SJR). Extension of the SJR boundary will provide a tool to investigate how the Delta may respond to different SJR management strategies.

The system domain for this project is the portion of the SJR from near Vernalis to the Mendota Pool (see Figure 3-1). The project was divided into two phases because of substantial gaps in bathymetry data. Phase I is that portion of the domain from the Bear Creek confluence near Stevinson down to the current boundary near Vernalis. Phase II is that portion of the domain from Stevinson to Mendota Pool. In general, the SJR boundary extension work reported herein is limited to Phase I.

3.2 Model Development

A set of USGS 7.5-minute topographic maps encompassing the project area was used to discretize the domain into 92 reaches with 93 nodes (Phase I & II). The locations of the nodes generally correspond to a hierarchy of major tributaries, possible point sources of inflow and outflow, or convenient landmarks. The geographic coordinate of each node was manually measured from the maps using the Universal Transverse Mercator, Zone 10 (UTM) reference system. The length of each reach was manually measured from the maps using a digital planimeter. Three values per reach were measured then averaged. The reaches are approximately 1-2 miles long.

Bathymetry data for the system domain were obtained from the U.S. Army Corps of Engineers (USACE). The data were transformed from the latitude/longitude coordinate system to the UTM coordinate system using "Corpscon," public domain software developed by USACE. The transformed bathymetry data and nodal coordinates were then input into the Department's Cross Section Development Program (CSDP).

CSDP was used to define the system geometry, such as channel alignment and cross sections, for input to DSM2. The model's river reaches were defined by aligning centerlines to follow the thalweg (low flow channel) that was visually located from the bathymetry data graphically displayed by CSDP. A new function was added to CSDP that calculates the reach length from the aligned centerlines. However, special care is

necessary for this function to give sufficient results. The thalweg can be difficult to visually extract from the data and is highly sinuous. The placement of many short centerline segments may be necessary to accurately define a meandering channel alignment. Many short segments were used to describe the channels in CSDP. As a benchmark, the reach lengths computed by CSDP were compared to the manual planimeter measurements. The net difference between the two methods was small (approximately two feet), with CSDP yielding the greater length.

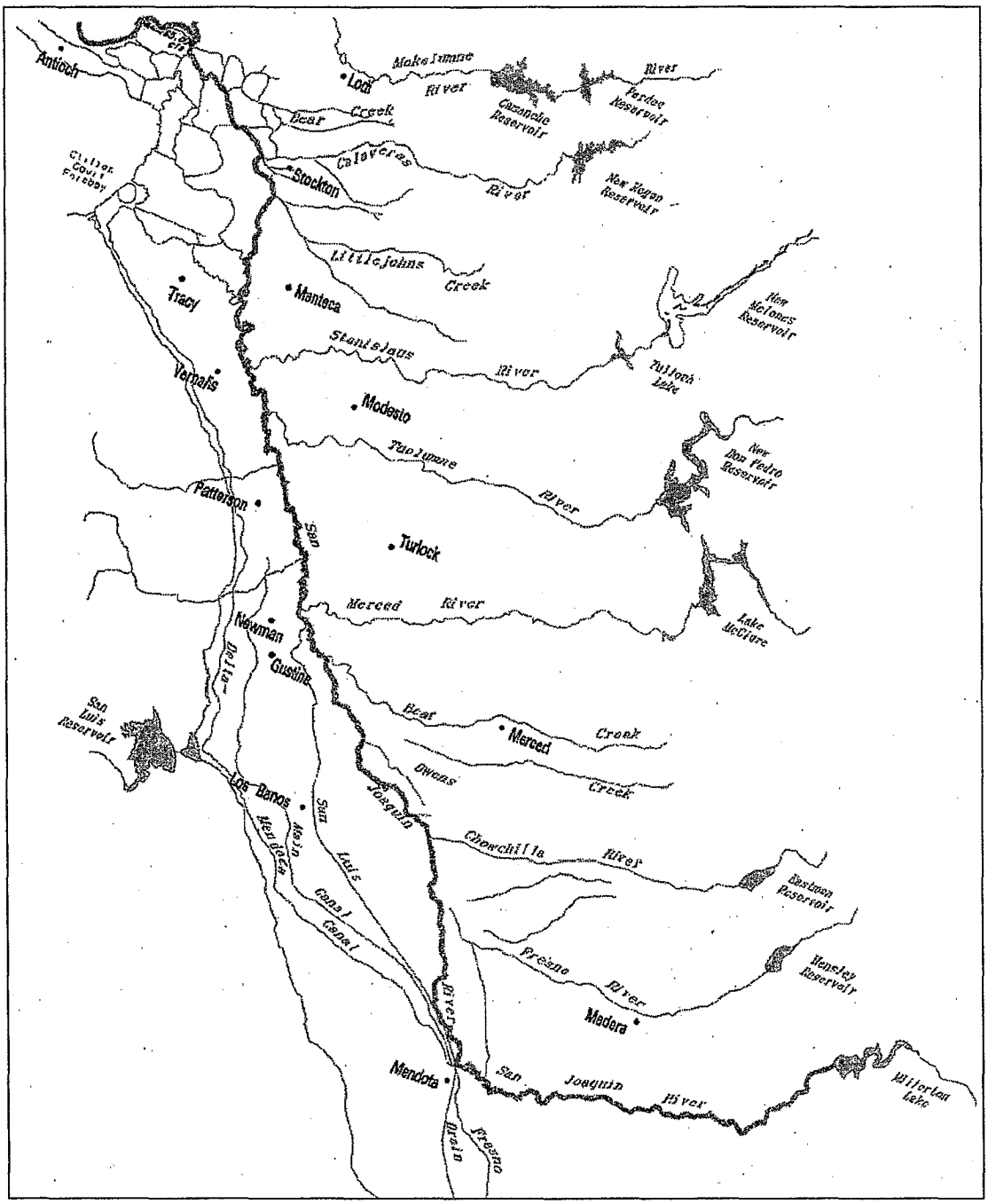

Figure 3-1: San Joaquin River.

Irregular cross sections were developed using CSDP to approximate the river's existing natural shape. Every channel has at least one representative irregular cross section and some have as many as three. Personal engineering judgement and ability to distinguish a realistic cross section from the data displayed at chosen locations determined the initial location of the cross sections within a channel. In most cases the thalweg of the cross section was well defined but the floodplain was not. Digital aerial photos were used to reasonably approximate the shape and extent of the flood plains.

Even with the use of irregular cross sections, DSM2 still requires the definition of two rectangular cross sections per channel segment. These rectangular cross sections are only used if there is not at least one irregular cross section in a given channel segment. Therefore, a homogeneous rectangular cross section width of 500 feet was specified at the upstream and downstream sides of each node with a linear bottom slope between nodes. The slope was calculated using the change in channel elevation from the upstream boundary near Stevinson to Vernalis, approximately 60 feet (msl) to 0 feet (msl), respectively, divided by the number of reaches between those locations. A stage of 12 feet above the bottom elevation was specified for the initial condition.

3.3 Pre-Calibration Model Runs

A mock planning study was developed for the first trial run of the model. The purpose of this exercise was to test the planning mode input files and new geometry for design flaws. A few select periods with hydrologic conditions representative of dry, normal and wet scenarios were chosen. The hydrology for the Delta and major SJR tributaries was obtained from the DWR Planning Simulation Model (DWRSIM). Agricultural consumptive use was not readily available for the SJR and was neglected for these preliminary simulations.

The major problem encountered in the first trial run was channels drying up for the dry hydrologic scenario. DSM2 will not allow a discontinuity in the flow regime and model calculations will not proceed if a channel dries up. This error can typically be attributed to large changes in cross sectional area or dramatic changes in bottom elevation between irregular cross sections.

A systematic approach was developed to debug the geometry design. The model was run until a channel segment dried up then the irregular cross section(s) with that channel segment was (were) removed and the model ran again. If no irregular cross sections are defined for a given channel segment, then the model will default to the rectangular cross section defined for that channel segment. This process was repeated until the model ran to a successful completion. Approximately 40 percent of the irregular cross sections were removed, most of them consecutive and localized to four general areas. This consecutive and highly localized trend suggested that not all of the cross sections removed were problematic.

The elevation of a default rectangular cross section in one channel segment may not closely match an irregular cross section in a neighboring channel. This requires the introduction of a continuous block of rectangular cross sections where the elevations of the upstream and downstream ends of this section approximate the elevations of the neighboring irregular cross sections. Also, a problematic cross section may not cause an error in its own channel segment but may cause an error in other channel segments in close proximity. In some cases where a channel segment had multiple cross sections, only one cross section was the source of error.

Based on these conclusions, a refinement process was conducted to differentiate potentially good cross sections from the problematic ones. Each problem area was investigated independently of the others. Irregular cross sections were reintroduced and removed in systematic combinations until only a minimal number of irregulars were necessary to be removed. This process reduced the number of likely problematic cross sections to approximately 35 percent.

The next step was to determine which cross section was the likely source of the problem for the group. Two visualization methods were applied. The first step was to plot a family of stage to cross sectional area relationship curves for a problematic cross section and a few upstream and downstream of that cross section (see Figure 3-2). The other was to sequentially plot the bottom elevations of a problematic cross section and a few neighboring cross sections (see Figure 3-3). These tools were valuable assets to determine which geometric attribute was most likely to be causing the problem. In all cases, the bottom elevation transition was found to be the problem. The model generally experienced channel drying with changes in elevation greater than 5 feet between cross sections, sometimes more or less depending on the horizontal distance between them.

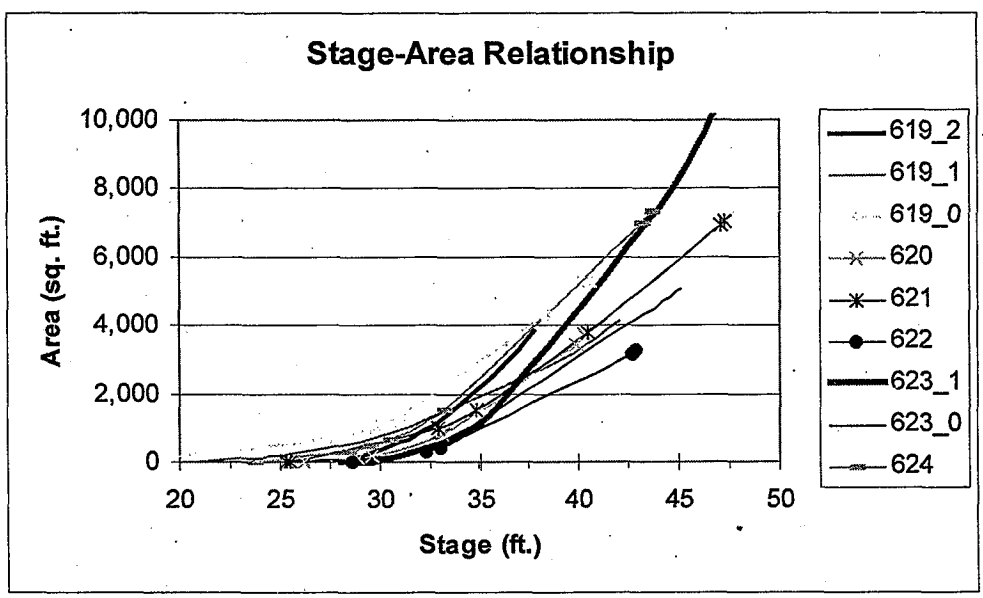

Figure 3-2: Stage-Area Relationship for channels 619 to 624.

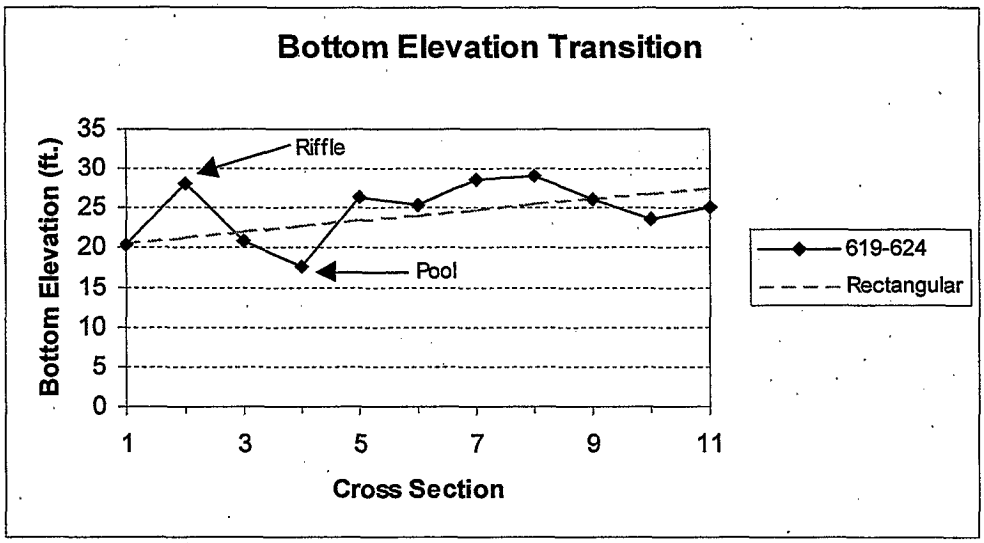

Figure 3-3: Bottom Elevation Transition for the Irregular Cross Sections from Channels 619 – 624.

Some of the deep poles and shallow riffles needed to be averaged. The bottom elevations of the corresponding rectangular cross sections were superimposed on a "bottom elevation transition" plot such as shown in Figure 3-3. This provided a reference baseline to a working slope since the model ran successfully when those cross sections were used as substitutes. The bathymetry data was revisited to determine a better location in the channel to draw a representative cross section with a bottom elevation closer to this baseline. In a few cases where a channel had more than one irregular cross section, a surplus section was deleted when relocation failed. After some iteration, the model ran to a successful completion without substitution of rectangular cross sections.

3.4 Future Directions

In coordination with the SJRMP's Water Quality Subcommittee,

1. Collect historical hydrology data and calibrate DSM2-HYDRO for the boundary extension;
2. Collect historical water quality data and calibrate DSM2-QUAL for the boundary extension; and
3. Complete Phase II.

4 VPlotter

4.1 Introduction

VISTA is a group of tools for retrieving, managing and manipulating time series data. Time series data is stored in HEC-DSS format. This allows efficient and easy access to large amounts of time series data.

VISTA provides the following four tools:

1. VISTA GUI. This is a graphical user interface that provides a fast and easy way of accessing time series data in HEC-DSS format.
2. VPLOTTER. This is a graphical user interface that bridges the gap between user interfaces and scripts by providing a template approach to repeatable operations.
3. VSCRIPT. This is a scripting interface for controlling the many individual components of VISTA.
4. VSERVER. This is a tool for distributing data over a network.

VPlotter is the latest in this group of tools for retrieval, manipulation and management of time series data. VPlotter is built on top of existing functionality of VISTA and VScript. Whereas VISTA stresses on browsing a DSS file or files, VPlotter stresses on repeatability and automation. VPlotter can call any function that is available using VScript and thus is as extensible as VScript. This chapter summarizes VPlotter. For details on VISTA, see the previous annual progress reports *Methodology for Flow and Salinity in the Sacramento-San Joaquin Delta and Suisun Marsh* (1998, 1999).

4.2 Installing VPlotter

The VISTA package comes with an installer which installs all the above mentioned programs. A user can download the zip file from the Delta Modeling Section's home page at `http://wwwdelmod.water.ca.gov`, unzip it and double click on the install.bat file. This starts a user interface that guides the user in installing the product. Once installed on Windows, a VISTA folder appears in the user's START button which contains the above mentioned tools. This VISTA folder contains all the VISTA tools, including VPlotter. VISTA is supported on Solaris and Windows platforms and with minimal effort can be ported to any platform supported by Java.

4.3 Using VPlotter

VPlotter sets up a template of operations to retrieve data from HEC-DSS files, manipulate and operate on that data, and then execute a set of standard actions on the resulting set of time series. This standard set of actions includes: Graph, Tabulate, Run Script, Write To DSS, Write To Text, Save To Web and Save To GIF/JPEG. Standard time series arithmetic operations are available with time series specific operations such as moving average and period averages. Scripts that may be written by users may provide additional operations.

VPlotter defines three levels: study, graph, and plot. The topmost level is the study level. The study level can contain multiple graphs. The graph level can contain multiple plots. This hierarchy is represented by a tree structure similar to a file manager. The selected level is highlighted and the operations are done with context to that level.

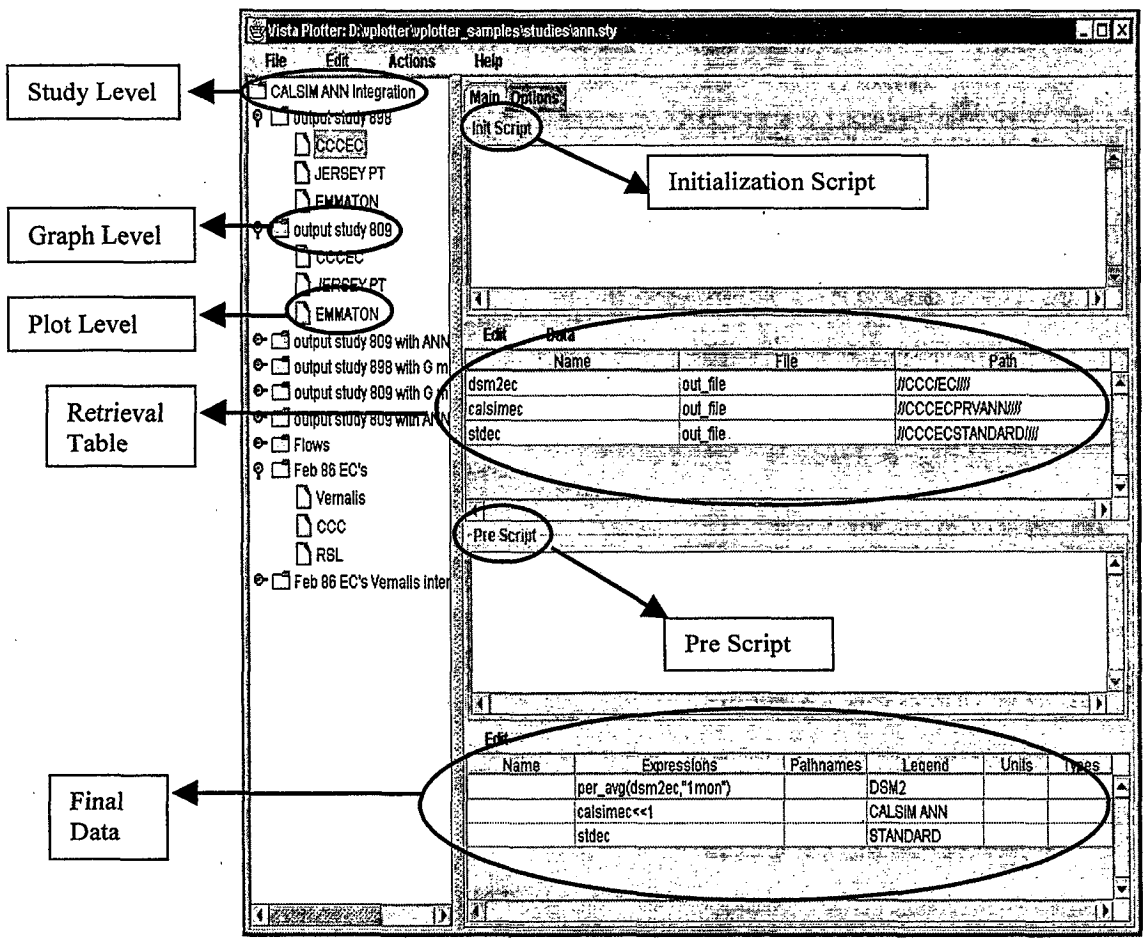

Figure 4-1: Sample Vplotter Session.

At the study level the user can define global definitions that are accessible at all contained levels. The graph level contains its own set of definitions as well as a graph title and the preferred layout of the plots in the graph.

The plot level contains its own set of definitions, a table of named items that are time series retrieved from a filename and pathname specification. This is followed by a set of actions called "Pre-script" and then a table with expressions that result in time series. These expressions result in time series on which the set of standard actions applies.

The options tab at the plot level contains the time window for retrieval, the plot title, the axis labels and their range. Arbitrary plot text is allowed and a comment section is allowed for each plot as well.

Users can extend VPlotter's functionality by writing functions in VScript. Existing time series can be examined and data can be extracted from them. This data may be used from multiple time series to create another set of data from which a new time series may be created. The created time series is identical in behavior to time series retrieved from HEC-DSS files and can be manipulated identically.

User help is available in the form of online help in VPlotter and VISTA. There is also documentation for advanced users, which is generated from the source code for VISTA. In addition, the underlying languages used by VISTA, which are Jpython and Java, are well documented at `http://www.jpython.org` and at `http://java.sun.com` respectively.

4.4 References

Sandhu, Nicky. (1998). "Chapter 8: VISTA (Visualization Tool and Analyzer)." *Methodology for Flow and Salinity Estimates in the Sacramento-San Joaquin Delta and Suisun Marsh. 19th Annual Progress Report to the State Water Resources Control Board.* California Department of Water Resources. Sacramento, CA.

Sandhu, Nicky. (1999). "Chapter 5: VSCRIPT: A Scripting Language Extension for VISTA." *Methodology for Flow and Salinity Estimates in the Sacramento-San Joaquin Delta and Suisun Marsh, 20th Annual Progress Report to the State Water Resources Control Board.* California Department of Water Resources. Sacramento, CA.

5 DSM2 Particle Tracking Model Development

5.1 Introduction

This chapter provides an update to further develop the DSM2 Particle Tracking Model. It covers the background of the model including the history, overview of the model and the basic underlying principals. The chapter describes the development and incorporation of particle behavior into the model and the Graphical User Interface to define these behaviors.

5.2 Background

5.2.1 History

In June 1992 the Department of Water Resources (DWR) hired Gilbert Bogle of Water Engineering and Modeling to develop a nonproprietary particle tracking module that DWR could adapt output and geometry of its DSM1 model. The original module provided by Dr. Bogle was a quasi two-dimensional model, which simulated longitudinal dispersion by utilizing a vertical velocity profile, vertical mixing, and a dispersion coefficient (which was a function of velocity, depth, and width of the channel). Because of the tidal nature of the water system and the channel grid, some complications occurred when implementing the module. After some scrutiny, the model was developed to be a quasi three-dimensional model.

The Particle Tracking Model was originally written in FORTRAN. Because of the model, the code was partially rewritten in C++ and Java to take advantage of using an object-oriented approach. The code was modified to handle the new output from the DSM2 model. The input system was rewritten in FORTRAN to use and to be consistent with the DSM2 input system.

5.2.2 Overview of Model

The Particle Tracking Model (DSM2-PTM) simulates the transport and fate of individual notional "particles" traveling throughout the Sacramento - San Joaquin Delta. The model uses velocity, flow and stage output from a one-dimensional hydrodynamic model (DSM2-HYDRO). Time intervals for these hydrodynamic values can vary but are on the

order of 15 minutes or 1 hour. Time varying input into the hydrodynamic model includes inflows at various rivers, exports, agricultural return and diversions, stage at Martinez, and gate operations. Fixed input includes static channel, reservoir and gate properties, and other runtime parameters.

The Delta's geometry is modeled as a network of channel segments and open water areas connected together by junctions. The PTM uses the same fixed input files that describe the geometry for DSM2-HYDRO. The particles move throughout the network under the influence of flows and random mixing effects.

The location of a particle at any time step within a channel is given by the channel segment number, the distance from the downstream end of the channel segment (x), the distance from the centerline of the channel (y), and the distance above the channel bottom (z).

5.3 Particle Movement

Three-dimensional movement of neutrally buoyant particles within channels is accomplished by longitudinal, transverse, and vertical movement.

5.3.1 Longitudinal Movement

The longitudinal distance traveled by a particle is equal to a combination of two velocity profiles multiplied by the time step. The two profiles are described in transverse and vertical directions.

5.3.1.1 Transverse Velocity Profile

The transverse profile simulates the effects of channel shear that occurs along the sides of a channel. The result is varying velocities across the width of the channel. The average cross-sectional velocity during a time step (Figure 5-1), which is supplied by DSM2-HYDRO, is adjusted by multiplying it by a factor which is dependent on the particle's transverse location in the channel. This results in a transverse velocity profile where the particles located closer to the shore move slower than those located near the centerline in the channel, as shown in Figure 5-2. The model uses a quartic function or fourth order polynomial to represent the velocity profile. The velocity profile is represented by the equation:

$$F_T(y) = A_q + B_q\left(2\frac{y}{w}\right)^2 + C_q\left(2\frac{y}{w}\right)^4$$

where:

A_q, B_q, C_q	=	Coefficients,
y	=	Distance across the channel from –0.5 to 0.5, and
w	=	Width of the channel.

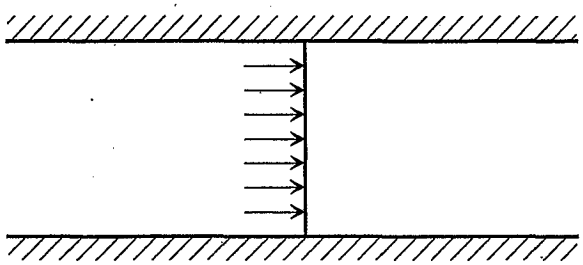

Figure 5-1: Average velocity over the transverse.

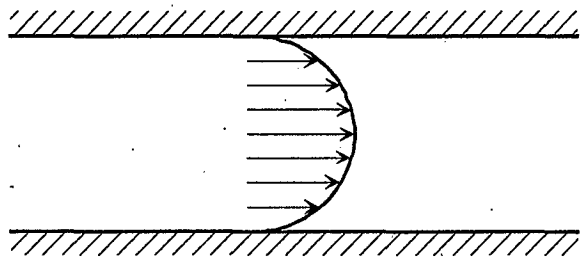

Figure 5-2: Applied quartic function to the average velocity in the transverse.

5.3.1.2 Vertical Velocity Profile

The average cross sectional velocity (Figure 5-3) is adjusted by multiplying it by a factor that is dependent on the particle's vertical location in the channel. This results in a vertical velocity profile where the particles located closer to the bottom of the channel move slower than particles located near the surface. The model uses the Von Karman logarithmic profile to create the velocity profile as shown in Figure 5-4.

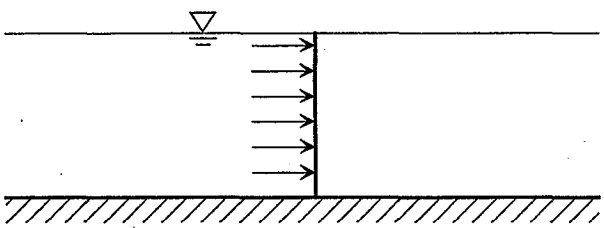

Figure 5-3: Average velocity over the vertical.

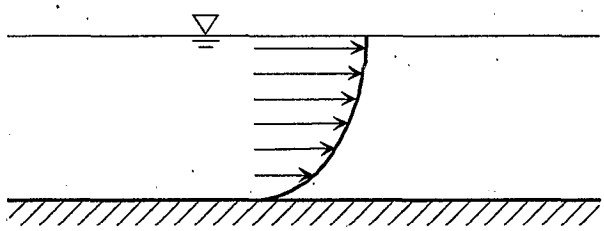

Figure 5-4: Applied Von Karman log function to the average velocity in the vertical.

5.3.2 Transverse Movement

Particles move horizontally across the channel because of mixing. A gaussian random factor, a transverse mixing coefficient, and the length of the time step are used in the calculation of the distance a particle will move during a time step. The mixing coefficient is a function of the water depth and the velocity in the channel. High velocities and deeper water result in greater mixing.

The transverse mixing is in the process of being re-vamped to include a more detailed calculation of the transverse mixing coefficient. The gradient of this coefficient will be included in the calculation to more accurately simulate dispersion. The gradient will introduce displacement resulting from the variation in velocity within the channel.

5.3.3 Vertical Movement

Particles also move vertically in the channel due to mixing. A gaussian random factor, a vertical mixing coefficient, and the length of the time step are used in the calculation of the distance a particle will move during a time step. The mixing coefficient is a function of the water depth and the velocity in the channel. As with transverse mixing, when there are high velocities and deeper water the mixing is greater.

The vertical mixing is in the process of being re-vamped to include a more detailed calculation of the vertical mixing coefficient. Again, the gradient of this coefficient will be added in to more accurately simulate dispersion.

5.4 Capabilities

- Particles can be inserted at any node location within the Delta.
- History of each particle's movement is available. In the model, the path each particle takes through the Delta is recorded. Output for determining the particle's movement includes:
  - Animation. Particles are shown moving through the Delta Channels. The effects of tides, inflows, barriers, and diversions on particles are seen at hourly time steps;
  - Number of particles passing locations. The number of particles that pass specified locations are counted at each time step; and
  - Number of particles within a specified group of channels and reservoirs.
- Each particle has a unique identity, and characteristics can change over time. Since each particle is individually tracked, characteristics can be assigned to the particle. Examples of characteristics are additional velocities that represent behavior (self-induced velocities) or the state of the particle such as age.
- Particles travel at different velocities at different locations within the cross section. The PTM takes the average one dimensional channel velocity from DSM2-HYDRO and creates a quasi three-dimensional velocity field. So, if particles are heavy and tend to sink toward the bottom, they will move slower than if they were neutrally buoyant. As a result, travel time through the channels is longer.

5.5 Behaviors

Until recently, PTM simulations primarily have been made using neutrally buoyant particles. Working with biologists within the IEP Estuary Ecological and Resident Fish Studies Project Work Teams, some behaviors have been incorporated into the model. Some studies have been made where settling velocities and mortality rates have been included. These studies have concentrated on Stripped Bass eggs and larvae. The introduction of behaviors to the PTM will allow for the simulation of other things. Additional behaviors have been added to restrict a particle's movement within a given volume. This restriction can be induced by the time of day or flow direction (in progress).

5.5.1 Life Stage or Phase

The idea of a *life stage* or *phase* is important to simulating with behavior. A *life stage* represents a particle's behavior at a given time. Figure 5-5 shows an example of the life stages of an imaginary species of fish. The figure shows three defined life stages. Each stage is given a name and a group of properties that make up the behavior. The first stage shown represents the behavior associated with an "Egg." The second stage shown represents the behavior associated with the "Larvae." The third stage shown represents the behavior associated with the "Juvenile." The *life stage* would be a definable period in the life of a particle, whether the particle relates to fish or a non-conservative dye.

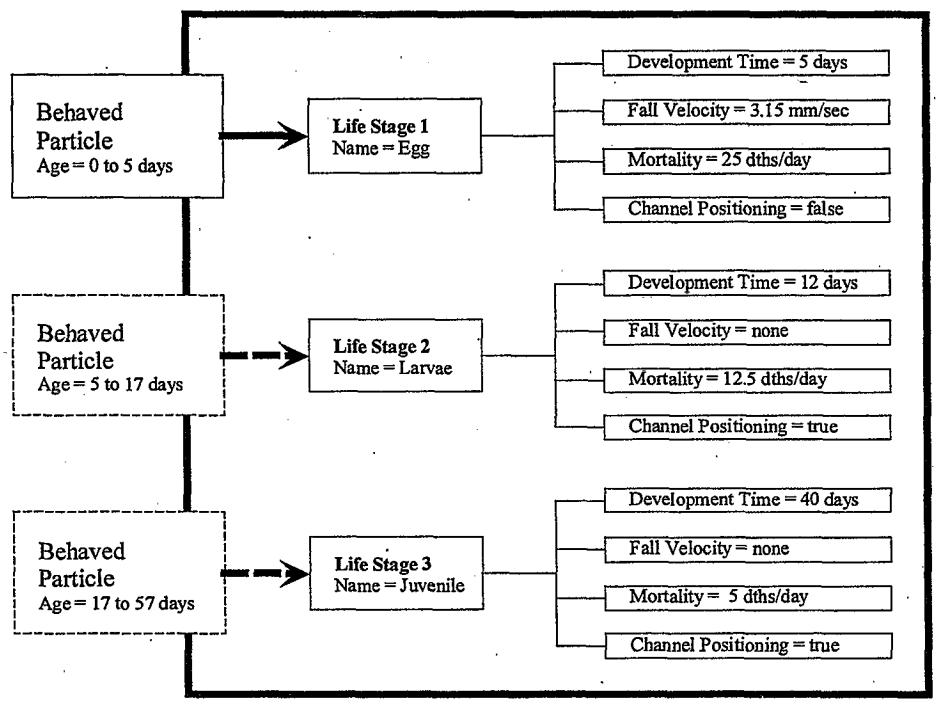

Figure 5-5: Diagram of particle Life Stages.

5.5.2 Development Time

A *life stage* is given a *development time* that defines the lifetime of a particular *life stage*. The *development time* tells the PTM how long to simulate a particle with a given behavior. Going back to the *life stages* of the imaginary fish (Figure 5-5), the *development time* of an "Egg" is defined to be five days. So for five days a particle will have the behavior of an "Egg." After five days the particle takes on the behavior associated with a "Larvae." Then 12 days later, 17 days total, the particle takes on the behavior of a "Juvenile."

When "particles" age, their behavior may change. For example, eggs may have a defined density and may float or sink and die. When the eggs turn into larvae they may not have a defined density, but they may swim a specific way.

5.5.3 Fall Velocity

A *fall velocity* can be added to a particle. This adds an additional downward (+) or upward (-) velocity component to a particle. This can be useful such as in simulating Stripped Bass eggs, where eggs are deposited at the top of the water column. The eggs have a slightly higher density and tend to fall at a predictable rate.

5.5.4 Mortality Rate

A *mortality rate* is the probability that a particle will die within a given time period. *Mortality rate* is determined using the exponential distribution:

$$P(X) = 1 - e^{-\lambda t}$$

where:	$P(X)$	=	Probability of death per time period,
	λ	=	Exponential distribution coefficient, and
	t	=	Percent time.

Given a probability of death per time period, λ is calculated by setting $t = 1$:

$$\lambda = -\ln(1 - P(X))$$

Particles for each *life stage* are assigned a uniform random number between 0 and 1. Particle death then occurs if it's random number is less than the current cutoff number. The cutoff number is calculated by:

$$cutOffNum = 1 - e^{(-\lambda age)}$$

where:	*cutOffNum*	=	Probability of death per time period,
	λ	=	Exponential distribution coefficient, and
	age	=	Age of particle for current *life stage*.

Suppose there is a group of three particles, each assigned a mortality or probability of death of 25 percent per day. Knowing this information calculation of λ occurs by:

$$\lambda = -\ln(1 - P(X))$$
$$\lambda = -\ln(1 - 0.25)$$
$$\lambda = 0.288$$

Now that λ has been calculated, a uniform random number is assigned to each particle:

Part1 = 0.308,
Part2 = 0.972, and
Part3 = 0.029.

Now as time is incremented, each particle is checked to determine if they die. For this example the time increments will be 0.5 day. The first increment results in the following *cutOffNum(s)*:

$$cutOffNum = 1 - e^{(-\lambda age)}$$
$$cutOffNum = 1 - e^{(-0.288)(0.5)}$$
$$cutOffNum = 0.134$$

A *cutOffNum* of 0.134 results in the death of particle Part3. Part3 has a number of 0.029, which is less than that of the *cutOffNum*. Repeating this procedure at the given time step of 0.5 days, it is found that Part1 dies after 1.5 days and Part2 dies after 12.5 days.

5.5.5 Vertical Positioning

Vertical positioning allows for defining a restriction on the particle vertical movement in the channel. Typically a particle is allowed to roam 100 percent of the channel depth. Figure 5-6 shows particles distributed throughout the water column. These particles can potentially be subjected to any portion of the velocity profile. With *vertical positioning* the particles are restricted within a defined range. As shown in Figure 5-7, the particles are restricted to the lower portion of the channel. The range can be restricted to any part of the channel. The range can even be defined for a given time. With the restriction, the particles are subjected to the lower portion of the velocity profile.

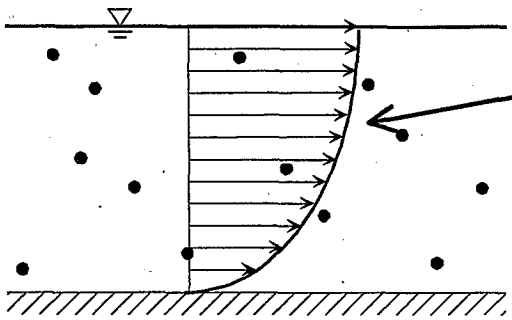

Figure 5-6: Normal particle unrestricted distribution.

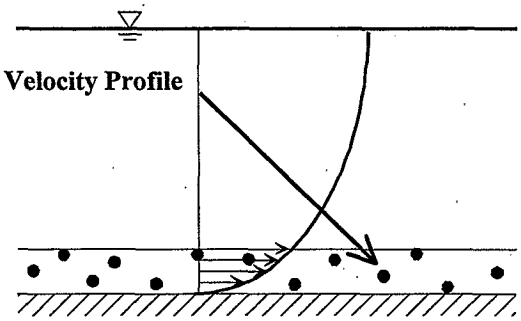

Figure 5-7: Particles restricted to lower portion of the channel.

5.6 Behaviors in Progress

There are a few other types of behaviors requested by the IEP Resident Fish Studies Project Work Team, which are in the process of implementation. The behaviors include transverse positioning, flow positioning and the incorporation of response to water quality.

5.6.1 Transverse Positioning

Transverse positioning is similar to the vertical positioning. However, with transverse positioning the range of restriction would be in the transverse or across the width. So particles can be forced to stay near the banks or near the center of channels. This could be useful if it is known that food exists at the sides of channels. A transverse positioning component can be included so particles can move toward the shore at certain times in the day. For example, particles may swim toward the shore for food.

5.6.2 Flow Positioning

Flow positioning would include a type of vertical and transverse positioning. The driving force for decisions would be the direction of the flow. Vertical and transverse positioning could be assigned for a given flow direction, whether seaward or landward. An example would be a particle moving up in the water column to ride the flood or ebbing tide and then move down in the water column to reduce its velocity and movement in the opposite direction.

Along with positioning, particles may include an additional velocity depending on the flow direction. Particles may have an additional longitudinal velocity component. Particles may swim with or against the flow.

5.6.3 Quality

The incorporation of the DSM2-QUAL output into the PTM will allow particles to react to many different water quality constituents, including (but not limited to) temperature, dissolved oxygen, pesticides, and food abundance. This development will allow a particle's growth rate and mortality to be a function of water quality. Particles could also swim toward certain water quality or food abundance.

5.7 Graphical User Interface

The Graphical User Interface (GUI) is currently available for editing particle behaviors. The following is a description of this behavior GUI.

5.7.1 Main Frame

When the GUI is first initialized, the user gets a frame like shown in Figure 5-8. New *life stages* can be added by using the text field and adding the name. Saved behaviors can also be loaded from this window. Figure 5-9 shows this window with *life stages* added. To illustrate the use of this tool, an "Egg" behavior and "Larvae" behavior will be used.

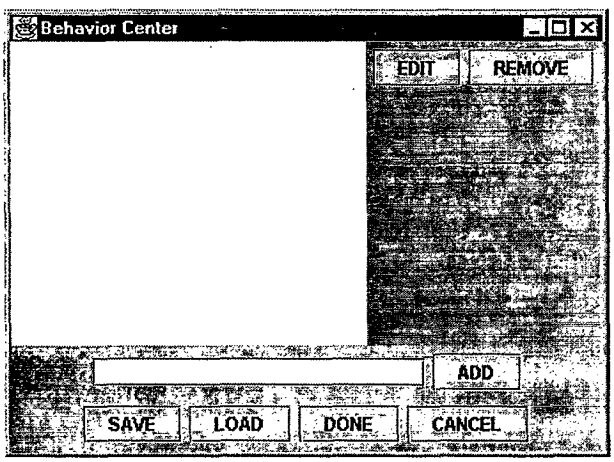

Figure 5-8: Initialized behavior GUI.

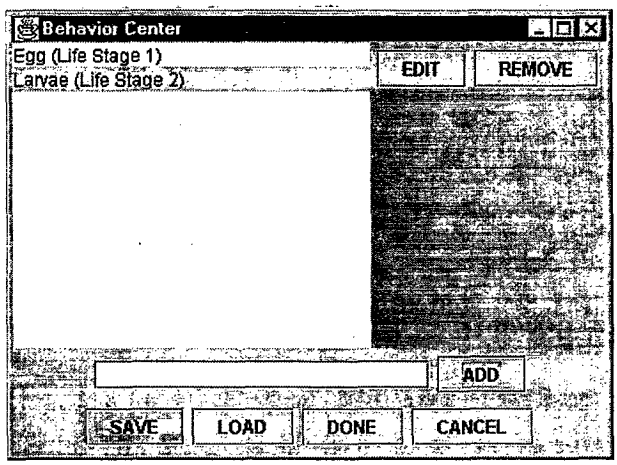

Figure 5-9: GUI with *life stages* entered.

5.7.2 Physical definitions

After either adding or loading and editing a *life stage*, a new window with a group of tabbed panes is displayed. Figure 5-10 shows the physical properties tab for the "Larvae (Life Stage 2)" where the physical properties of the *life stages* are defined. The *development time* is defined here as 10 days long. There is no *fall velocity* with this *life*

stage so it is left undefined and assumes a value of zero. The *mortality rate* or probability of death is set to 10 deaths per day.

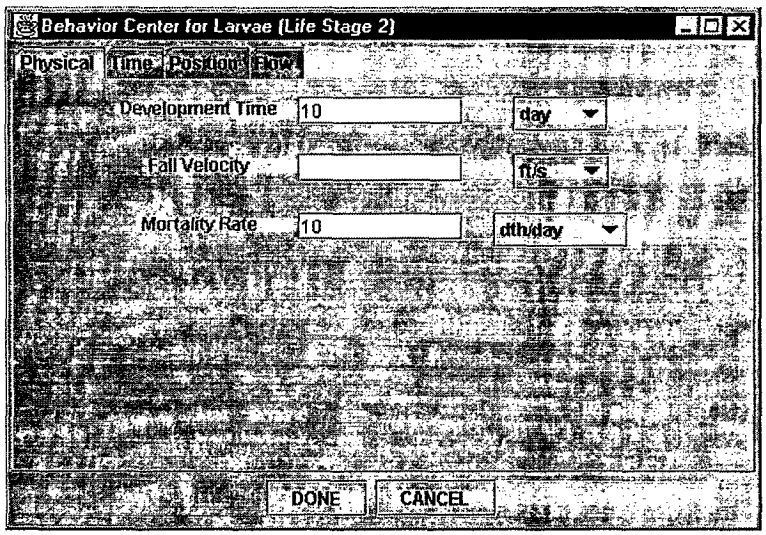

Figure 5-10: Physical properties tab.

5.7.3 Channel Positioning

The positioning tab allows the user to define areas in the channels that the particles must stay within. Figure 5-11 shows the positioning defined for "Larvae (Life Stage 2)." There is currently a limit of four vertical and horizontal channel positions for each *life stage*. The first column defines the limit of the lower percent channel depth. The second column defines the limit of the upper percent channel depth. The third and fourth columns define the start and ending times for the particular positioning.

In the example shown in Figure 5-11, the particles during this *life stage* are forced to the upper 10 percent of the channel depth (from a lower limit of 90 percent depth to an upper limit of 100 percent depth) for the time between 6 a.m. and 6 p.m. The particles are then forced to the bottom 10 percent of the channel depth for the time between 6 p.m. and 6 a.m.

All times do not need to be defined. For any undefined time during the day, the model defaults to fully mixed.

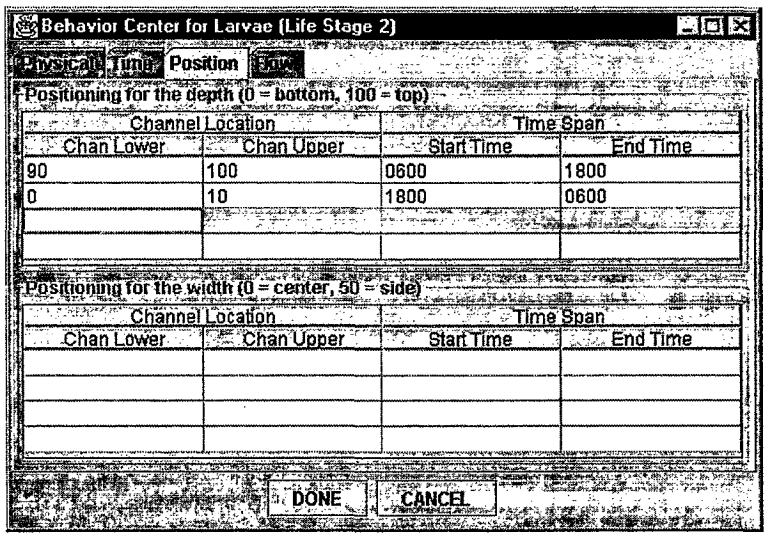

Figure 5-11: Channel positioning tab.

Returning to the main GUI and saving this information there creates a file. The format of the file utilizes the eXtensible Markup Language (XML), so it can easily be edited by hand. This file is then specified in the general DSM2 input file structure.

5.7.4 Future Graphical User Interface

A full PTM interface is in development. This will allow for the setup and execution of model simulations through a graphical interface. The interface will display results of studies along with providing particle animations.

5.8 Additional PTM Enhancements

Some other PTM enhancements include some *scalar* flags, and the channel group output.

5.8.1 Channel Groups

Channel groups allow for the grouping of channels and reservoirs. The model keeps track of and outputs the number of particles in the defined groups. This type of output is useful in looking at the distribution of particles at a given time. Definition of these groups is done through a text-input file using the keyword *GROUP*. The fields for the groups are *object*, *number*, and *group*. The *object* field defines a channel with "chan" or reservoir with "res." The *number* defines the object number (the channel or reservoir number). The *group* field defines the group in which this object should be included.

The following example shows how to define a group. For this example channel 275 and reservoir 5 have been associated with group number 2.

```
GROUP
OBJECT      NUMBER      GROUP
# big break
chan        275         2
res         5           2
END
```

5.8.2 Scalar Flags

A few new scalar flags have been added to increase the flexibility in the PTM output and the transverse velocity behavior. As with typical DSM2 scalar logical flag behavior, a true value corresponds to the value indicated in the variable name.

ptm_flux_percent -	Logical flag allows flux output as a percent or actual values.
ptm_flux_cumulative -	Logical flag allows flux output as cumulative or instantaneous values.
ptm_group_percent -	Logical flag allows group output as a percent or actual values.
ptm_no_animated -	Integer flag specifies the number of particles to track for purposes of animation.
ptm_trans_a_coef -	Floating point flag specifies the "a" coefficient in the quartic function used to describe the transverse velocity profile.
ptm_trans_b_coef -	Floating point flag specifies the "b" coefficient in the quartic function used to describe the transverse velocity profile.
ptm_trans_c_coef -	Floating point flag specifies the "c" coefficient in the quartic function used to describe the transverse velocity profile.

6 Effects of Salinity-Induced Density Variation on DSM2 Hydrodynamics

6.1 Introduction

With DSM2, the Delta is modeled assuming a one-way coupling between hydrodynamics and salinity. Flow and stage results from HYDRO are used to drive the salinity transport model QUAL, but ignore feedback that salinity-induced density gradients have on hydrodynamics. The assumption of no feedback is made for convenience, but is supported by order-of-magnitude arguments that play down the importance of baroclinic terms in the governing equations. DSM2 does not support a coupled solution of hydrodynamics and transport.

Numerical testing of the effects of variable density is a task that is revisited from time to time. Recent motivation for investigation came from the 1999-2000 IEP PWT calibration project, where certain mild systematic errors were thought to be attributable to density. This report documents a simple numerical experiment on the importance of longitudinal density variation on 1-D hydrodynamics in DSM2 (no representation is made that this captures all the nuances of baroclinic hydrodynamics).

The experiment was from September 15, 1997 to October 31, 1998. The original idea was to span two IEP calibration periods, one in April 1998 and another in September 1998, plus a warm-up period for the quality model. Salinity varied considerably over this period, as demonstrated by the tidally averaged (EC) plot of Figure 6-1. However, the most important high salinity periods occurred before the first IEP calibration periods. The April 1998 calibration period had almost zero salinity, and in the September 1998 period salinity was low to medium.

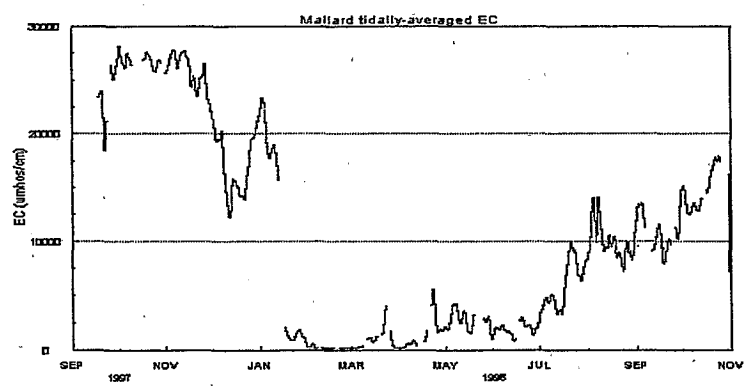

Figure 6-1: EC at Mallard (RSAC075) during the study period.

6.2 Methods

Internally, DSM2 is able to incorporate variable density into the flow equations -- this capability is a legacy of the FourPt model written by Lew de Long. However, there is no supporting input/output system for inserting density information and DSM2 does not solve the full hydrodynamic and salt transport system simultaneously.

For the present project, EC output from QUAL was used to calculate a density estimate offline, which was then reinserted into HYDRO using a crude input system designed for this experiment.

Three model runs were carried out for the experiment:
1. a preliminary HYDRO run to establish a flow field;
2. a QUAL run to estimate salinity; and
3. a follow-up HYDRO run.

The feedback cycle could, of course, be repeated *ad nauseum* with successively improved flow fields passing from HYDRO to QUAL, and improved density fields returning from QUAL to HYDRO. The scheme is not guaranteed to converge, but due to the small changes over one cycle it seems likely to do so very rapidly. Such detail was not required here, and if density effects were to be incorporated formally into DSM2, the iterative process would probably be replaced by simultaneous solution of hydrodynamics and salt transport.

Density was estimated by converting EC to total dissolved solids (TDS) and then adding the TDS concentration to the density of water to obtain the density in solution. The conversion from EC to TDS was calculated using regression results from the Suisun Marsh Reports (`http://iep.water.ca.gov/suisun/`). A single conversion formula for Mallard Island (RSAC075) was applied over the whole delta:

$$\text{TDS (mg/l)} = -60.06 + 0.614 * \text{EC (umhos/cm)}$$

The range of density over space and time was between 1.0 g/cc and 1.022 g/cc, the latter is about two-thirds of the value quoted by Fischer (1972) for seawater. The variable density results were compared to a base case with fixed density – none of the results were compared to field observations.

6.3 Results

The main difference between the "base case" (with constant density) and the "variable density case" is that stage in the variable density case is up to several tenths of a foot higher. Figure 6-2 shows stage at Mallard (RSAC075) and Figure 6-3 shows flow on the Sacramento near Sherman Lake (RSAC084) for a typical time segment during the early part of the experiment, when salinity was high. Proportionally, the difference in stage is larger than the difference in flow.

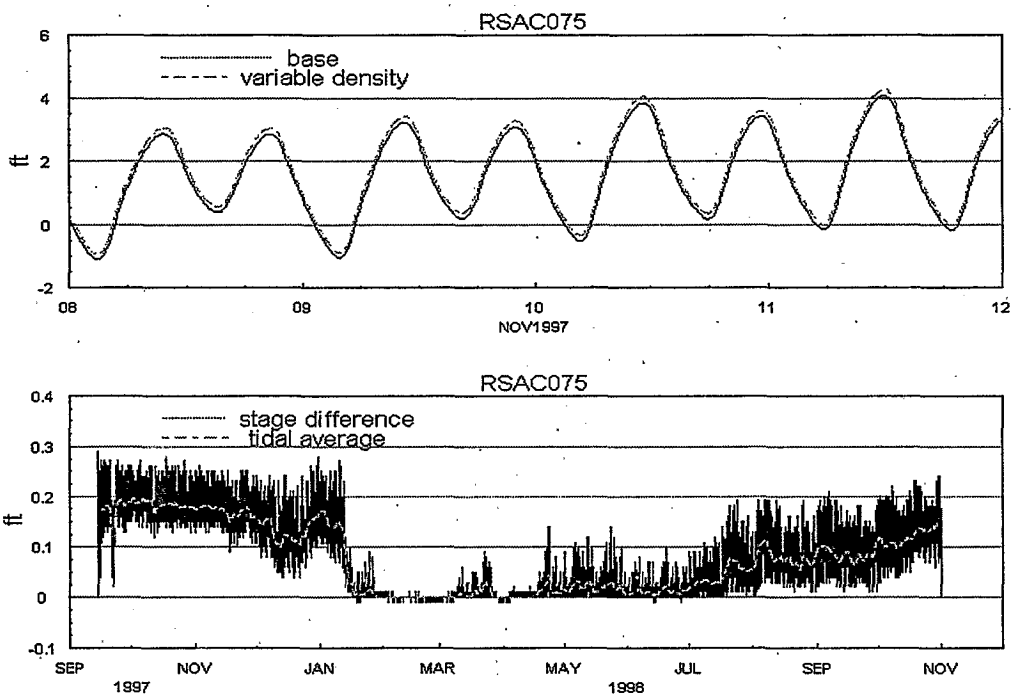

Figure 6-2: (Top) Stage results for Mallard (RSAC075). (Bottom) Difference between the two cases (variable density minus base) and its tidal average.

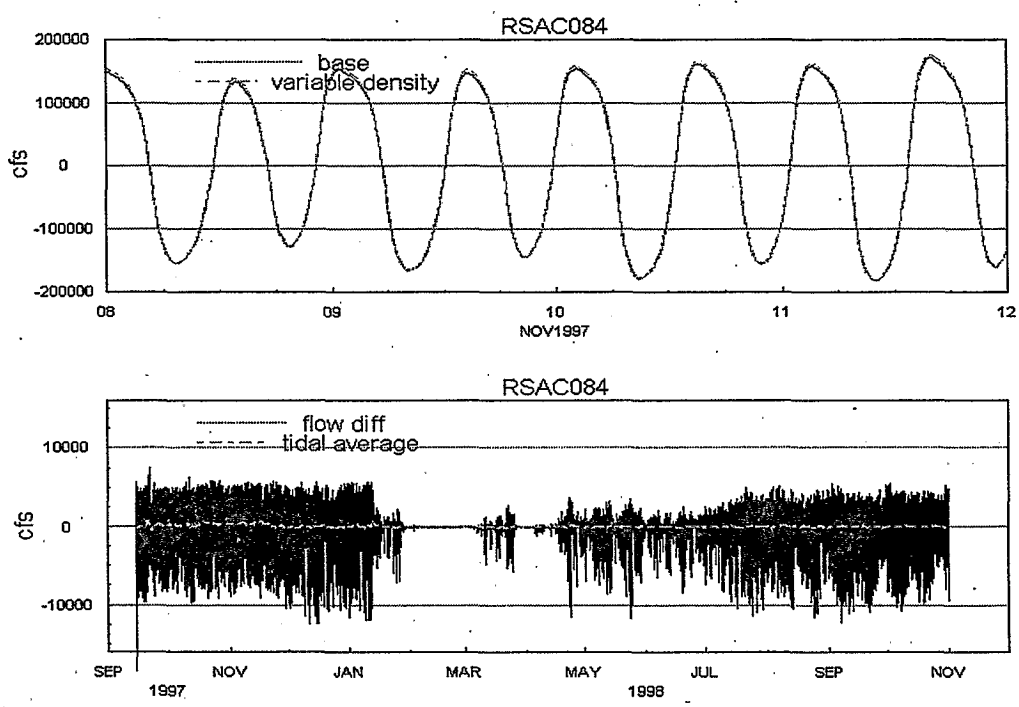

Figure 6-3: (Top) Flow near Sherman Lake. (Bottom) Difference between cases (variable density minus base) and its tidal average.

The right side of Figure 6-2 shows the evolution of stage and flow differences during the experiment (variable density minus base). This difference varies tidally, and a filtered (tidally averaged) line has been added to give some idea of the "average" difference between the two cases. Not unexpectedly, the difference between the two cases depends on how much salt is in the Delta (compare to Figure 6-1).

The stage differences between the base and variable density cases are fairly uniform over the Delta. Figure 6-4 shows stage difference on the Sacramento at Collinsville (RSAC081), Rio Vista (RSAC101), Delta Cross Channel (RSAC128), and at the head of Old River (ROLD074). The amount of tidal variation is different from location to location, but the trends are the same and have a similar magnitude.

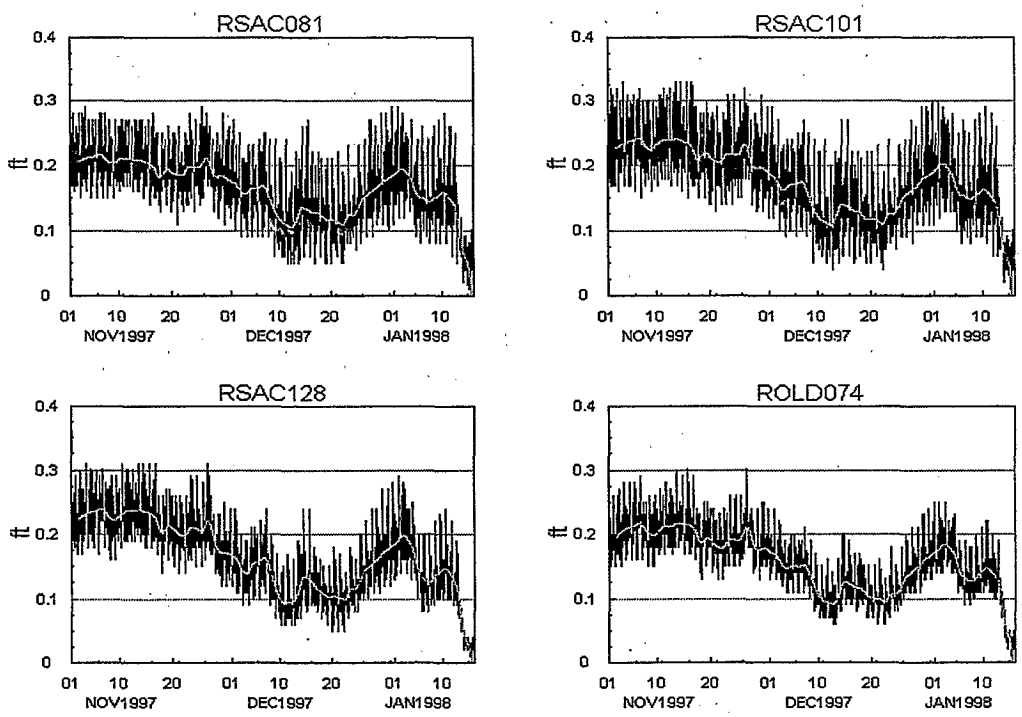

Figure 6-4: Stage difference (variable density minus base) at four locations: Collinsville (RSAC081), Rio Vista (RSAC101), Delta Cross Channel (RSAC128), and Old River at Head (ROLD074).

The lack of spatial variation is not surprising. Density is highest and the density gradient steepest near Martinez –magnitude and gradient decrease precipitously more than 20-30 km toward the east and south. West of Emmaton or Jersey Point the density gradients induced by salinity are always gentle and insufficient to produce hydrodynamic change on their own. Instead, the changes that are induced in most of the Delta are just due to the propagation inward of changes near the western boundary. It is as if the "boundary condition" had been changed.

6.4 Conclusions

The inclusion of density gradients influences model results. Whether the magnitude of the effect is important will depend on application, but it can be called "mild" compared to other sources of model error. Baroclinic effects depend strongly on the timing of hydrology (they are almost nil when salinity is low), but only weakly on location. Because the IEP calibration periods in 1998 are times of low salinity, longitudinal baroclinic effects were probably not the cause of the errors that motivated this experiment.

6.5 Reference

Fischer, H.B., E.J. List, R.C.Y. Koh, J. Imberger and N.H. Brooks. (1972). *Mixing in Inland and Coastal Waters*, Academic Press, San Diego.

7 Artificial Neural Networks

7.1 Introduction

Artificial Neural Networks (ANNs) are able to approximate any function to any degree of accuracy given enough internal nodes (Sandhu and Finch, 1996). ANNs run significantly faster than the functions they approximate with only a small loss of accuracy. CALSIM has the capability of using ANNs (as well as the G-Model) to approximate Delta flow-salinity relationships. The ANNs used are created from the Stuttgart Neural Network Simulator (SNNS). More recently, ANNs have been used to generate salinity values at Martinez for planning and forecasting studies.

During the last year, numerous approaches were taken to improve the accuracy of the ANNs. These approaches included:

- The use of different node network structures;
- Perturbing the values of the inputs to reduce correlation and increase the amount of available data; and
- Refining the scripts that generate the ANNs to improve their stability.

7.2 Different Node Structures

The current network structure of the ANN consists of 72 input nodes. Each of these nodes connect to a layer of eight internal hidden nodes that connect to a second layer of two internal hidden nodes and end with a single output node.

The 72 values entered into the input nodes come from 4 input sources, each of which have 18 input values. The four input sources are the Delta Cross Channel position, amount of combined exports and diversions, Sacramento River inflow and San Joaquin River inflow. Sacramento inflow includes Sacramento River flow, Yolo Bypass flow, and the combined flow from the Mokelumne and Cosumnes Rivers (also known as the East Side Streams or ESS). The combined exports and diversions includes: the State Water Project (SWP), Central Valley Project (CVP), and North Bay Aqueduct exports, Contra Costa Water District diversions, and the net channel depletions. These net channel depletions are calculated using the Delta Island Consumptive Use (DICU) model. The 18 input values for each input source come from the 8 daily values followed by 10 weekly averages. These values cover the previous 77 days worth of values and the current daily value. These 78 days of values create a memory that allow the ANN to

account for the influence of the previous state of the system and calculate the state of the system.

The two layers of internal hidden nodes allow for interactions between the input nodes. By allowing the input nodes to interact, the normal interactions that would occur from the inputs of the system can be approximated. The weights that get assigned during training reflect the amount of influence that different inputs have on each other. The output node yields the final result that comes from the application of all of the weightings on the inputs.

One network that was experimented with increased the number of internal nodes in the first internal hidden layer to 16 and the second internal hidden layer to 4. This configuration increased the time to train, but did not significantly increase the accuracy of the results. The introduction of additional internal nodes allows for more flexibility of the network and as such allows for the ANN to approximate more spikes. At a certain level of increased nodes, the network will lose integrity and become unresponsive due to not having enough data to properly condition the nodes. By increasing the amount of data, the number of internal nodes that can be supported is increased.

Another network experiment involved splitting the first hidden layer into five hidden nodes. Two of the nodes contained linkages to only the northern inputs: the Delta Cross Channel (DCC) position and Sacramento River flow. Two more of the nodes contained linkages to only the southern inputs: the combined exports and diversions and San Joaquin River flow. The last node contained linkages to all four sources. This was tested because the sources within the northern and southern areas are more likely to have an impact on each other prior to interacting with the other area. This configuration increased the level of complexity of the ANN structure without significantly increasing the accuracy of the results.

7.3 Perturbation of Inputs

In an effort to increase model sensitivity to the effects of the individual sources and increase the amount of available data to train the networks, the inputs to the ANN were perturbed. Traditionally, while training the ANN, the effect of the DCC position is isolated by using its normal operation and then an inverted open/closed schedule. This is done by running DSM2 for the normal operation and then running it again with the inverted operation and then using the inputs and outputs for both DSM2 runs for training the network. The desired effect of this perturbation is to isolate the effects of the DCC from the Sacramento River and remove their correlation to each other.

The network is normally trained on 5840 daily values retrieved from a standard 16-year DSM2-QUAL planning study from October 1, 1975 through September 30, 1991. The run is started at October 1, 1974, but the first year's worth of data (except for 77 days used for memory building) is discarded to account for the warm-up period that DSM2-QUAL requires. Although QUAL's warm-up period is only 3-4 months, the 1974

simulation data is not included in the training data set in order to keep the number of data points for each month uniform. Each perturbation increases the number of available training values by 5840.

In the same manner as with the DCC position, the remaining three input values were perturbed by 20 percent. Increasing and decreasing the values of each individual input while keeping the remaining inputs constant and using the normal DCC operation yielded an additional six 16-year DSM2 runs worth of values. The values for these six runs were added to the normal two done for the DCC for a total of eight runs worth of data. In addition to creating additional data for training, this data better reflected the influences of the individual inputs. The increase or decrease of the value for each input removed some of the correlation otherwise present in the training data when the inputs were not treated independently.

The additional data allowed for an increase in the number of internal nodes, from 8 to 32, without collapsing the integrity of the network. The increased number of nodes allowed the network to have more flexibility and increased the number of peaks that could be approximated.

7.4 Example of Using the ANN to Forecast Martinez Salinity

DSM2-QUAL requires that water quality conditions be specified at the downstream boundary located at Martinez (RSAC054). ANNs have been used by the Section over the past few years to estimate Martinez EC for DSM2 planning studies. This year a new ANN was developed to estimate Martinez EC for DSM2 forecasting applications.

Since this tool was designed for forecasting rather than planning applications, the training inputs were limited to data that would be readily available for forecasts. The ANN was trained on the following daily average historical data measured from May 19, 1983 through December 10, 1999: Martinez EC, Sacramento River flow at Freeport, San Joaquin River flow at Vernalis, and combined SWP and CVP exports. Smaller Delta inflows such as East Side Stream and Yolo Bypass are assumed to be small constant values in DSM2 forecasting applications; therefore, they were not included in ANN training. Delta Cross Channel operation was not included in ANN training as it has no measurable impact on Martinez salinity.

The training period was selected based on data availability. Missing values were filled in using a simple linear fit. The first three-quarters of the historical data were used for calibration and the last quarter was used for validation.

The trained ANN was first used in several DSM2-QUAL simulations of historical Delta conditions during November and December 1999. In these post-cast simulations, historical SWP and CVP exports were altered in order to study the impacts of project operations on Delta water quality. (Five different export scenarios were evaluated.) Export modifications have a direct influence on net Delta outflow and influence salinity

at Martinez – thus requiring estimates from the ANN. The trained ANN was applied in several DSM2-QUAL forecast applications conducted in January and February 2000.

The use of the ANN to forecast EC at Martinez has been replaced by a new boundary salinity forecasting process. This process is described in Chapter 11.

7.5 Conglomeration of Scripts

The scripts used to generate an Artificial Neural Network were written in a combination of Python, Awk and C-shell scripts. Recently all of the scripts were moved to a single scripting language (Python) and incorporated as a part of the new VPlotter program. Through this change, the generation of ANNs is now a more stable and repeatable process that is easier to perform.

7.6 Reference

Sandhu, Nicky and R. Finch. (1996). *"Application of Artificial Neural Networks to the Sacramento-San Joaquin Delta."* Estuarine and Coastal Modeling, Proceedings of the 4th International Conference.

8 Filling In and Forecasting DSM2 Tidal Boundary Stage

8.1 Introduction

An adequate characterization of stage at Martinez is critical in order for DSM2 to simulate delta dynamics accurately. This section discusses a method of modeling Martinez stage, applicable to short-term prediction and the filling of historical records. The model combines a traditional astronomical tide with a model of the residual tide (the part of the observed tide that is not explained by an astronomical model).

8.2 Data and Preliminary Observations

A comparison of the bay tides with astronomical tides provides an interesting introduction to the boundary stage estimation problem. Figure 8-1 shows the tide at four stations: San Francisco Presidio, San Pablo Bay (RSAC045), Martinez (RSAC054) and Mallard (RSAC075). A harmonic mean tide has been fit to each of the stations:

$$z(t) = \sum_{i=1}^{N} f_i a_i \cos(\omega_i t + \phi_i + u_i) \qquad (0.1)$$

where $z(t)$ is the harmonic mean tide, ω_i are known frequencies associated with astronomical motion, ϕ_i is a local phase, and f_i and u_i are slowly varying amplitude and phase adjustments attributable to the 19-year-cycle of the lunar node, as tabulated by the National Ocean Service (see Schureman, 1941).

The choice and the number N of constituents came from the standard NOS menu of 37 common constituents. In accordance with traditional NOS practice, constituents that had estimated amplitudes smaller than 0.03 feet were dropped – this may err toward over-fit, which is acceptable because the predictors are orthogonal and any "extra" coefficient estimates do not tend to hurt the "significant" ones. The final selection was slightly different for each station – between 21 (RSAC075) and 28 constituents (RSAC054) were retained per station. Many common constituents were used in all of the fits (the most important were M_2, K_1, S_2, N_2, O_1, Mf, (L2), Ssa, $2MK_3$, and $2SM_2$). The differences were in minor species and shallow water constituents, all of which had small amplitudes a few hundredths above the "cutoff" of 0.03 ft. Fewer species could be fit to RSAC075 than to RSAC054, despite RSAC075 being higher in the estuary where more shallow

water species have had a chance to develop. This suggests that distortion and noise play a limiting role on harmonic fits higher up in the estuary.

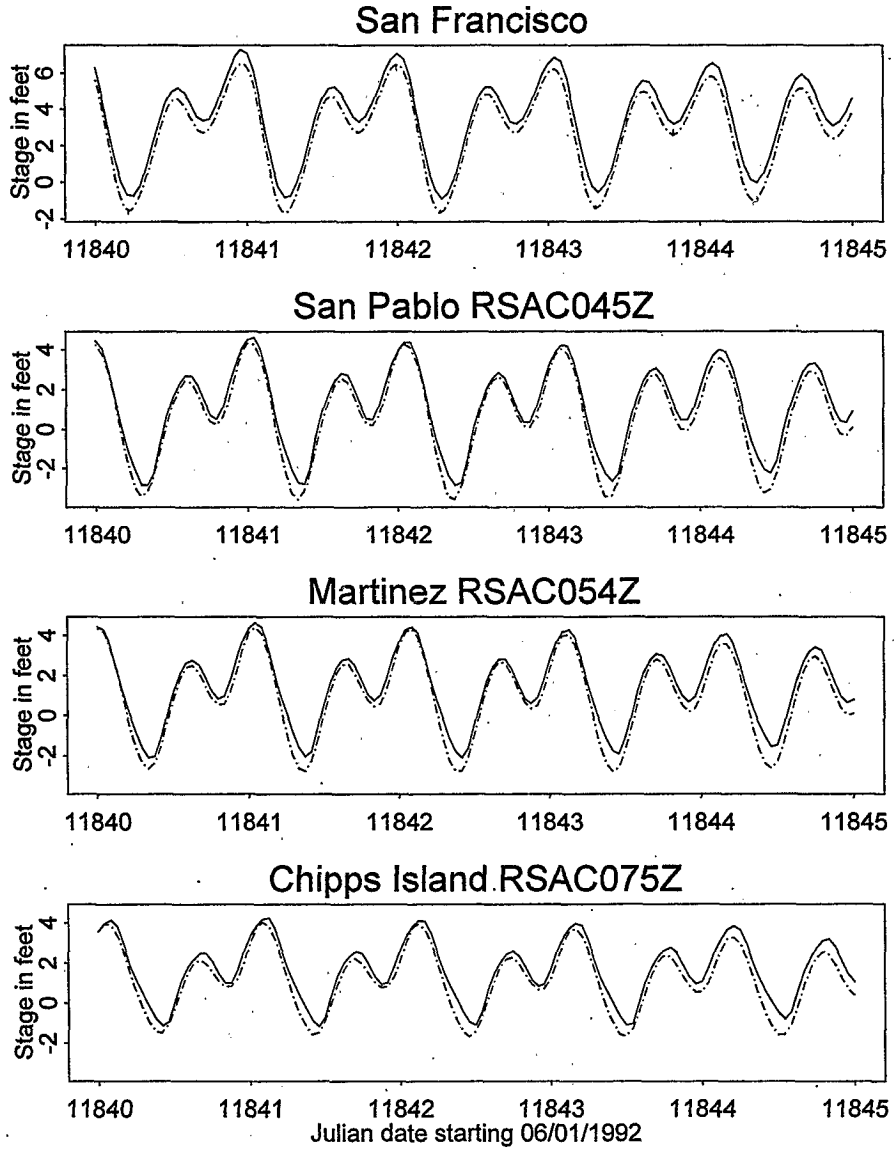

Figure 8-1: Tides (solid) and astronomical fits (dashed) at four stations.

Figure 8-2 (a) compares observed tide to astronomical over a single tidal cycle. Figure 8-2 (b) shows the astronomical residue over a fourteen-day cycle beginning on the same day, and Figure 8-3 compares the residuals at four stations. Two features are striking. First, there are local "trends" with period from several days to several weeks, often with magnitudes of a foot or more. On the right plot of Figure 8-2 long-wave variation is about one foot up over ten days (which is not an extreme example). The main physical contributors to these trends are barometric changes and 14-29 day oscillations induced by the interaction between lunar and solar constituents.

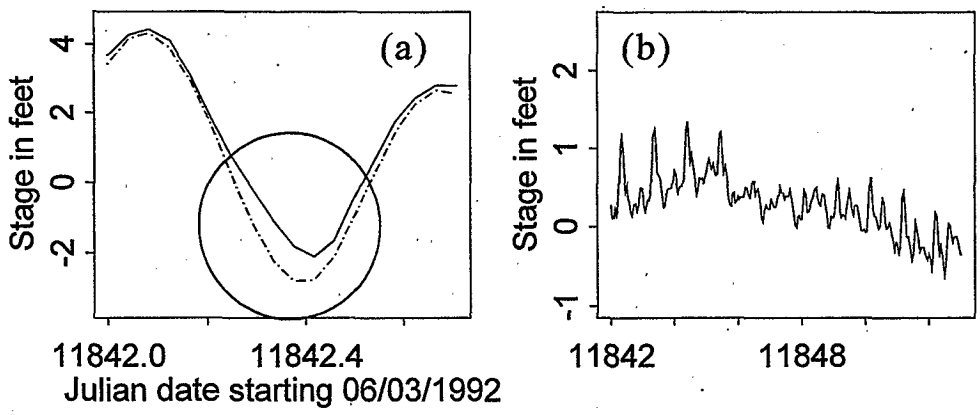

Figure 8-2: (a) Tidal distortion (solid observed vs. dashed astronomic) at Martinez over one tide cycle; (b) tidal residue during a 14-day spring-neap cycle.

The second source of error is that the tide is distorted (from sinusoidal) as it travels upstream, because of bottom friction, circulation in bays and effects induced by the shape of the basin. The episodes of greatest distortion cause instantaneous errors of about one foot at Martinez, and the complexity of the error develops as the tide moves upstream. Distortion is often manifest as patterns or "slivers" between the harmonic and observed tides (as in the circle inset in Figure 8-2), which last for several days. Common patterns are periodic, but recalcitrant to deterministic modeling – the patterns sometimes disappear and reappear for months at a time under circumstances which are sometimes not easy to tie to an external cause (season, delta outflow, pressure etc.).

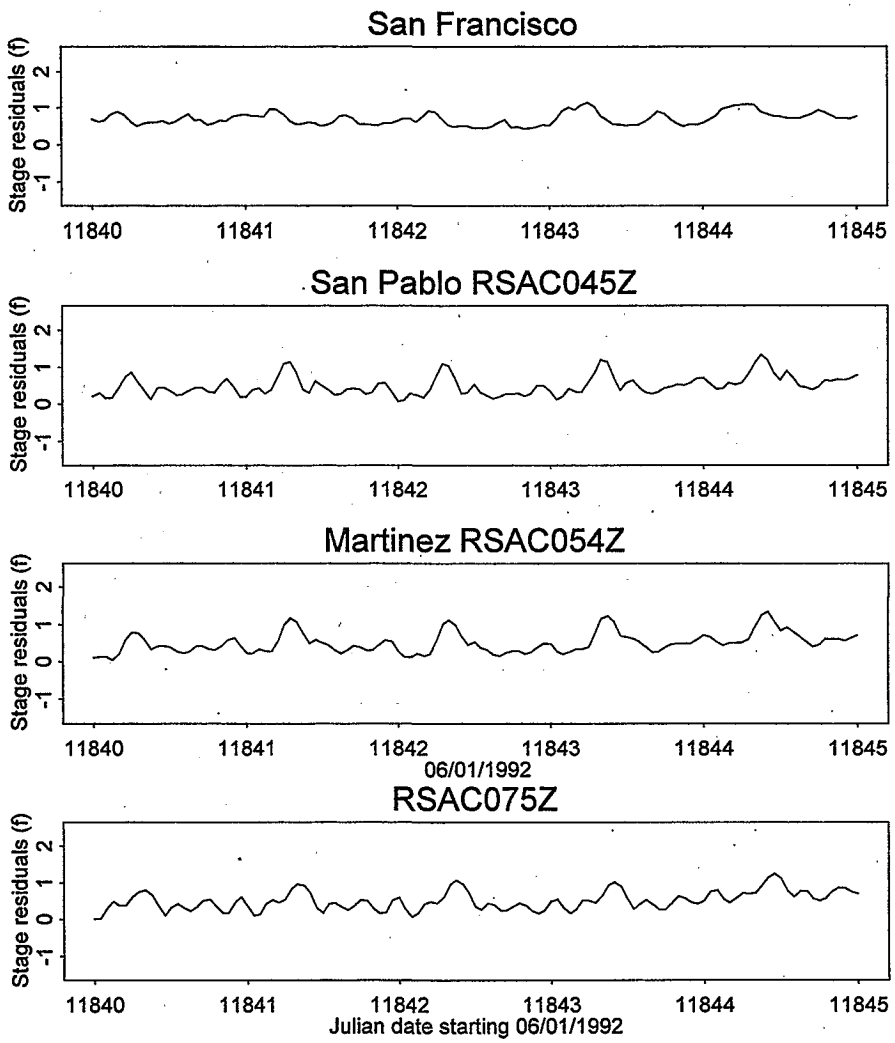

Figure 8-3: Residual tide at four stations (stage minus astronomical estimate).

8.3 *VAR Tide Residue Model*

8.3.1 Background in Linear Tide Models

The simplest method of linking tide observations at several stations is the linear model or convolution filter. Munk and Cartwright (1966) used linear (and quadratic Volterra-type) filtrations of an astronomical forcing function to produce tide predictions, and the same approach could be adapted to model one station on another in the estuary. The time domain representation of an appropriate filter model can be lengthy. For their application, Munk and Cartwright used a filter with lags spaced at $\Delta t = 2$ days in a two-sided filter spanning from -6 to +6 days, (thirteen terms). Their design was based on "equilibrium" tides filtered into narrow tidal bands. If their reasoning and spacing

criteria were applied to the full tidal spectrum (say 0 to 3 cycles/day), 4-hour spacing would be required over thirteen days.

Coefficient error and the short-term nature of shallow water distortion make a filter of such size unattractive for the present application. In a model of Martinez tide on San Francisco or San Pablo tide, 1-2 hour spacing over a maximum of 27 hours yielded much better fit (based on a Bayes Information Criterion) than did longer or more widely spaced filters. The only really important terms seem to be a few lags in the immediate past and another couple from the previous tidal day. Unidirectional filtration (modeling upstream stations on downstream) worked better than spatially two-sided filters, despite some bi-directionality in the underlying physical processes.

Removing the astronomic tide from the original tidal series improves the performance of the filtering approach and allows the use of a shorter filter. For instance, the best linear filter model of Martinez on San Francisco has a standard error of 0.27 feet if the full series is used; if instead residuals are used, the error is only 0.20 feet. It has been argued that with a true linear system, the decision as to whether to pass the mean tide through the filter or remove it and add it back later is arbitrary. This argument, neglects the effects of estimation error in determining the filter. When the mean tide is removed beforehand (eliminating line spectra), a much less complicated filter is required for the residuals and thus fewer filter coefficients have to be estimated.

For the remainder of this report, the astronomical tide will be taken to be an exogenous predictor – in other words it has already been computed before the rest of the analysis is carried out. An alternate approach would be to estimate the astronomical and residual components simultaneously. It is the author's experience from tinkering with the model that the residual model is not particularly sensitive to the astronomical model, and this at least is partial, informal confirmation of exogeneity. More detailed is deferred to later work.

8.3.2 Vector Autoregressive Model

Having made the above preliminary arguments, it is possible to specify a model that performs well. The model is a vector autoregressive (VAR) model based on astronomical residuals.

Because distortion appears to develop in an upstream direction, each station is modeled on its own past plus that of its downstream neighbor (except San Francisco, which is purely autoregressive), i.e.:

$$
\begin{aligned}
z_1(t) &= c_{1,1,1}\, z_1(t-1) + c_{1,1,2} z_1(t-2) + \ldots + c_{1,1,27} z_1(t-27) + \varepsilon_1(t) \\
z_2(t) &= c_{2,1,1} z_1(t-1) + \ldots + c_{2,1,27} z_1(t-27) + c_{2,2,1} z_2(t-1) + \ldots + c_{2,2,27} z_2(t-27) + \varepsilon_2(t) \\
&\ldots \qquad \ldots \\
z_4(t) &= c_{4,3,1} z_3(t-1) + \ldots + c_{4,3,27} z_3(t-27) + c_{4,4,1} z_4(t-1) + \ldots + c_{4,4,27} z_4(t-27) + \varepsilon_4(t)
\end{aligned}
\tag{0.2}
$$

where the $z_i(t), i = 1 \ldots 4$, respectively, represent water levels at the four stations San Francisco, San Pablo Bay (near the Carquinez Bridge), Martinez and Chipps Island/Pittsburg and the $c_{i,j,k}$ are coefficients in equation i of the jth station at lag k. The error terms ε_i are assumed to be Normal, to have a covariance matrix $\mathbf{Q}$, which includes correlation between stations but is independent and invariant over time (adequacy of the Normal model will be assessed shortly). The formulation of Equation (0.2) is general, but in practice only subsets of lags 1, 2, 3, 24, 25, 26, and 27 hours were found to be useful (using the Bayes Information Criterion, or BIC).

Coefficient estimates were obtained using Gaussian maximum likelihood, conditional on the first 27 hours of data. Given a complete tidal record, the likelihood calculations are equivalent to regressing each station on its own past lags and those of its downstream neighbor; i.e. fitting Equation (0.2) by least squares. The matrix $\mathbf{Q}$ is estimated by:

$$
\hat{\mathbf{Q}} = \frac{1}{T} \sum_{t=1}^{T} \hat{\boldsymbol{\varepsilon}}(t) \hat{\boldsymbol{\varepsilon}}(t)^{\mathrm{T}}
\tag{0.3}
$$

where the $\hat{\varepsilon}(t)$ represent one-step prediction errors.

Coefficient estimates are given in Table 8-1. The errors estimated from the constituent regression calculations were three to four orders of magnitude smaller than the coefficients. These regression estimates are not valid error estimates for the problem, but suggest that the coefficient estimates are "tight" and that significance is not an issue for the lags employed. Formal estimates of coefficient error were not made because such estimates are not a natural product of the EM algorithm (see below). The models have changed slightly in real-time applications because RSAC045 in the San Pablo Bay is no longer monitored.

lag	S.Francisco on self	S. Pablo SF terms	San Pablo	Martinez San Pablo	Martinez	Chipps Is. Martinez	Chipps Is.
1	0.713	0.303	0.722	0.233	0.854	0.442	0.890
2	0.203		-0.047		-0.144	-0.118	-0.322
3							0.108
24	0.302		0.255		0.290		0.291
25	0.186		0.398		0.366	-0.185	0.346
26	-0.206	-0.208	-0.394	-0.225	-0.439	-0.124	-0.395
27	-0.219		-0.034		0.060		0.072

Table 8-1: VAR model coefficients.

8.3.3 Missing Data

Missing data complicates the likelihood calculation (the fairly simple least squared problem described above). A significant fraction of the historical records at the tidal stations are missing – about 15-25 percent of the observations at San Pablo Bay, Martinez and Pittsburg. Since each "squared error" expression in the likelihood involves several stations and lags, any one of which could be missing, the fraction of expressions affected by missing data is very high indeed. Fortunately, many of the data are "missing at random," which means that the reason the data are missing is administrative or mechanical and is not directly related to the quantity being measured.

This section outlines how the missing data were treated using the EM algorithm. The EM (Expectation-Maximization) algorithm is an iterative numerical method for Maximum Likelihood Estimation (MLE), in which the original data are supplemented by "added" data or variables to form a "complete" data set. The added data are chosen in such a way that calculation of the likelihood using the complete data is easy. The added "data" are often in contrived forms.

The EM algorithm iterates between two steps:

E: calculate the expected value of the "complete" (log) likelihood expression using parameter estimates from the last iteration (or a starting value). Often, the likelihood expression is linear in the "added" data, in which case the E-step is equivalent to:

i. using the model to calculate the expected values of the "added" data necessary to "complete" the data set, and

ii. inserting the expected value of the "added" data into the expression for the likelihood.

M: maximize the "complete" likelihood expression from the E-step with respect to the parameters. In the VAR model this is just least-squares for the coefficients plus the estimate of $\mathbf{Q}$, as described above.

The naïve implementation of this idea would be to use the VAR to estimate missing stage residue observations, use the estimates to "complete" the historical record, and then re-estimate the VAR model. This turns out not to be an EM algorithm or even a good idea. Why? The likelihood is *not* linear in the raw stage values, but rather in sums of squares and cross-products – terms that look like $z_i z_j$ (i and j don't have to be distinct). In standard regression terminology, these are the terms usually denoted as $\mathbf{X}^T\mathbf{y}$ (or $\mathbf{X}^T\mathbf{Y}$) and $\mathbf{X}^T\mathbf{X}$. See Mardia (1979) for the applicable equations or Hamilton (1994) for the application of the likelihood to VAR models. Because the likelihood is not linear in the tide residue, substitution of even a good, unbiased estimate of the tide residue into the likelihood does not give an unbiased estimate of the likelihood.

Instead, we should choose our "added" data to be the missing cross-products. Here, "missing cross-products" refers to products that are needed for the likelihood but cannot be calculated because they involve at least one piece of missing data. The problem, then, is to calculate the expected value of cross-products.

In the case where both contributors to the product are missing, recall this basic identity for covariance:

$$E(z_i z_j) = E(z_i)E(z_j) + \mathrm{Cov}(z_i z_j) \tag{0.4}$$

where i and j represent times or stations and are not necessarily distinct, in which the covariance is a variance. The left side of Equation (0.4) is the "expected value of the product," which is what we want. The first term on the right side is the "product of the expected values." The second term on the right is the covariance between the two observations. This is the term that would be neglected under the "naïve" scheme. The covariance term tends to be particularly important for cross-products involving neighboring stations and time steps. This is usually the case. Cross-products arising from the least-squares fit of Equation (0.2) involve highly correlated lags and stations, and the missing data often occur in groups involving contiguous blocks of time or stations administered by the same agency.

In the simpler case, when cross-products involving one missing piece of data, the equivalent expression is:

$$E(z_i z_j) = E(z_i)z_j \tag{0.5}$$

Since z_j is known, it just acts as a scalar. In this case covariance does not come into play and the expression is linear in $E(z_i)$.

The expectations of cross-products can be calculated using Equation (0.4) or (0.5) using stage estimates and covariance estimates. To get both, the VAR model is put in state space form, and a Kalman filter and Kalman (Rauch-Tung-Striebel) smoother are used to produce the stage observations, cross-covariance and squared error terms conditional on all the data. Since the Kalman filter/smoother is used implementing the VAR model for historical filling, the EM algorithm can be employed without production of tools that would be of no further use once the parameter estimation is finished. The Kalman filter is not discussed further here – it is a computational device discussed in detail in any book on linear systems or state space modeling. See, for instance, Harvey (1994) or Hamilton (1994). The details of using the Kalman filter to obtain cross-covariance terms for use in EM are discussed in Watson and Engle (1983).

The use of the EM algorithm makes the VAR fitting complicated, and this process was never automated. Should the VAR model be refit later due to a change in circumstances (e.g., a change in the list of available stations), it would probably be adequate to use a least squares fit to Equation (0.2) plus Equation (0.3) using whatever data are available. It is certainly better to just throw away time steps whose expressions involve missing data at any lag than to implement an *ad hoc* iterative scheme in which the covariance between observations is ignored. As long as the data are missing at random, throwing out missing data still leads to an asymptotically efficient estimator.

8.3.4 Diagnostics

Rudimentary diagnostics were carried out to assess the "whiteness" and Gaussianity of the innovations process of the VAR model. The empirical autocovariance at Martinez of one-step prediction errors is shown at the top of Figure 8-4, along with a 95 percent confidence interval around zero (assuming a stationary Gaussian process). Residual periodicity is evident, but is much smaller than that of alternative filter lengths and spacing considered.

The bottom of Figure 8-4 is a distributional (q-q normal) plot based on 4,600 one-step prediction errors; there is little evidence of severe skewness or thick tails which would indicate that a normal distribution (and least squares) is inappropriate. Nevertheless, Chi-squared (p-value 0.0038) and two-sided Kolmogorov-Smirnoff tests (p-value of 5.0E-4) reject the hypothesis that the residuals are Gaussian, even after trimming a few outliers. Such results are typical because of the high power of goodness-of-fit tests based on large data sets – with so much data it is easy to prove that the data are not perfectly normal.

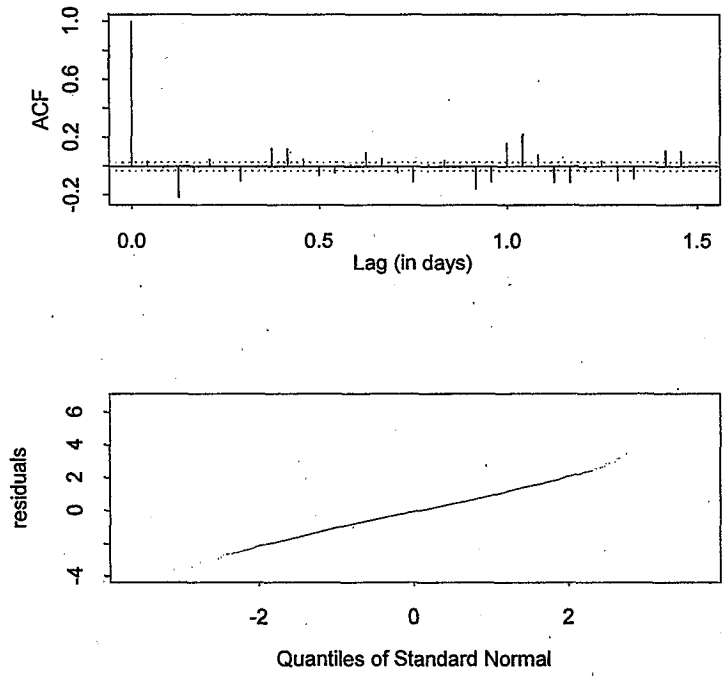

Figure 8-4: Diagnostics of the innovations process at Martinez. (Top) Empirical autocovariance. (Bottom) Quantile comparison with normal distribution.

8.3.5 Tide Series Reconstruction and Interpolation

After the residue has been predicted or filled, the final step is to add back the astronomical tide. The VAR component of the model (the part that works on the residue) operates on a 1-hour time step. To recover a 15-minute series for use in DSM2, the steps are as follows:

1. Interpolate the 1-hour estimate of the residue to 15-minutes, and
2. Add the result back to a 15-minute astronomical forcing.

Notice that only the "small" residual part is interpolated between the hours. The rest of the "shape" comes from the astronomical prediction, which can be resolved at any time step desired. Interpolating the hourly tide in this order is accurate, causing about ~0.01 foot error between the hour. Two alternatives would be to reconstruct the series first at a 1-hour time step and then interpolate the whole series, or to just reconstruct it at 1-hour and let DSM2 do the interpolating later. When linear interpolation is used, these alternatives are equivalent and result in errors at peaks of more than 0.1 foot, which is enough to spoil the accuracy of historical filling. Higher order interpolation is possible off-line, but not in DSM2.

In the production version of these tools, a fourth-order monotonicity-preserving spline due to Huynh (1993) was used for the interpolation, resulting in a more accurate and more visually pleasing series.

8.4 Applications

8.4.1 Prediction

The primary role of the VAR model in real-time modeling is to predict future stage. Predictions are produced by recursive application of the VAR Equations (0.2) to the tide residue with the error term set to zero. In practice, the synthesis of the VAR model is carried out with a Kalman filter, which is capable of producing variance estimates as well as a mean forecast. The Kalman filter is also required as the part of the historical filling algorithm. See Harvey (1994) or Hamilton (1994) for a discussion of how a vector autoregressive model is expressed in state space form for implementation in a Kalman filter.

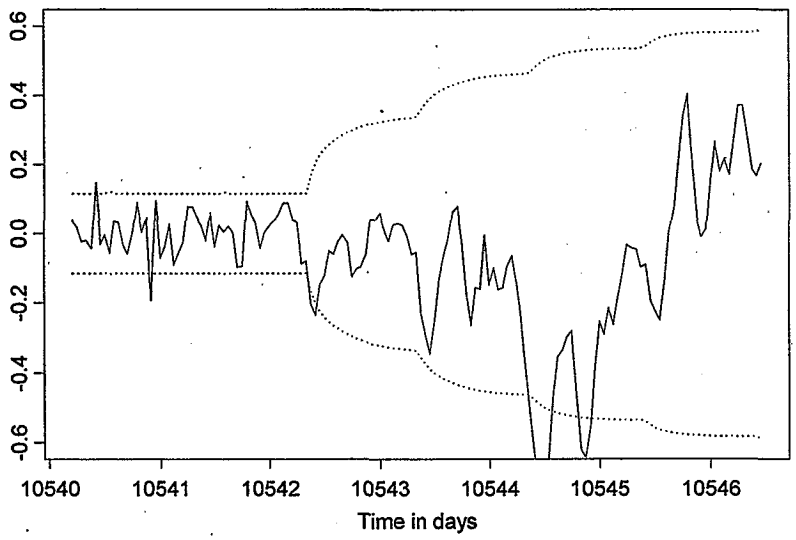

Figure 8-5: Errors and 95 percent confidence bound for simulated prediction using the VAR residue model (note sign: expected minus observed).

Figure 8-5 shows the results of simulated short-term forecasting at Martinez. All station records were artificially terminated at Julian date 10541 (November 10 1988-89). Until this point, the prediction errors and 95 percent confidence bounds represent one-step prediction error; after the cutoff date, the prediction error accumulates. Every tidal day, a discontinuous increase occurs in the confidence bounds, as real observations are replaced by estimates in the lagged components of the model. The confidence region is point-

wise: the probability of error continuously following the envelope is much smaller. The errors are reversed in sign from the usual for residuals (they are model minus observed).

The prediction error shown in Figure 8-5 takes on its worst values in the middle of Julian day 10,544. This can be explained in terms of long-period "trend" on that day. Figure 8-6 shows tidally averaged stage for the period. An unanticipated event occurred several days into the prediction period, causing water surface elevations to rise. This event is not captured in the VAR model, which can "remember" events that are in progress at the time of prediction, but cannot anticipate new barometric events. The attenuation portion of the event is captured well by the VAR model (as is a period of further attenuation not pictured). Further consideration of barometric events is given shortly; this example – a new event beginning soon within the prediction period – will be cited as a "tough case".

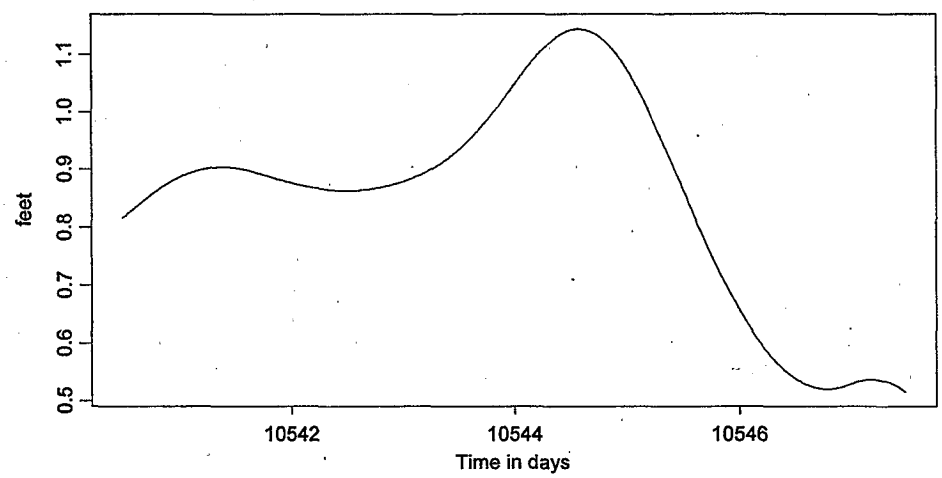

Figure 8-6: Tidally filtered stage during the simulated prediction period.

8.4.2 Filling missing records

The ability to fill historical records is a strong point of the VAR model (it was originally designed for this purpose), but is a task of only secondary importance in real time modeling. What distinguishes the filling (smoothing) problem is that supporting information is available from other stations. Since low-frequency fluctuations are felt fairly uniformly over the whole delta, the algorithm does well even during extreme events.

Although the same VAR formulation is used for prediction and filling missing records, the filling problem involves more complicated "smoothing" calculations. This is because we want to condition our estimate on all data available, which will include data before and after the historical gap. A single recursion marching forward in time is not sufficient, because later observations will not be used to improve earlier estimates. Instead, we need to make use of a bi-directional algorithm, and the one used here is a forward Kalman

filter coupled with a fixed-interval backward Kalman smoother (see Harvey, 1994). The result is an estimate for each missing value based on all available data.

Figure 8-7 shows errors resulting from two examples of simulated data filling, in which artificial gaps were created in the Martinez record. In series (a), which comes from the same period as the prediction problem of last section, Martinez goes off-line for two days. During the middle of this period San Pablo also goes off-line for several hours, causing a bump in the confidence interval. Note that the event that caused an episode of high error in the prediction problem is not even detectable in the filling problem.

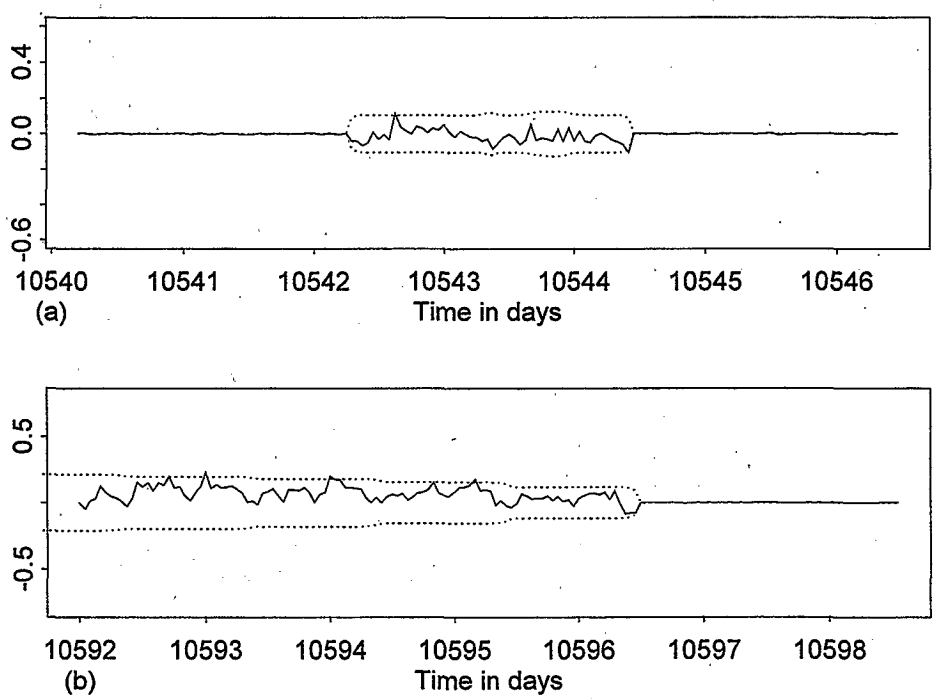

Figure 8-7: Errors for the smoothing of simulated gaps, plus 95 percent confidence bounds.

The main difficulty in filling data for real-time use seems to be the lack of supporting station data. Station RSAC045 was recently dismantled or moved. The lack of supporting station data will not matter much for small gaps, when the auto-regressive Martinez component does most of the work, but will make the model less accurate for long gaps. The San Francisco Presidio station is always available, and can be relied upon to pick up long wave events such as storms, but does not include information about shallow water distortion higher in the estuary.

Series (b) is a synthetic gap preceded by a month-long real gap in the Martinez record, and thus the model operates without autoregressive information from this station. San Pablo was also off-line at the beginning of the plot, coming on-line just before Julian day

10,595. Even during the worst long gaps, the smoothed Martinez record has a standard error of less than one tenth of a foot and a 95 percent confidence bound of about two-tenths of a foot. This level of accuracy has been verified with cross-validation over a large number of artificial gaps (every other week in 1996-1997), including episodes of moderate to high winds.

8.5 Discussion

8.5.1 Atmospheric Events

Correspondents who inquire about the stage boundary model are usually concerned with performance during atmospheric anomalies. The VAR model does not explicitly include barometric input, though the addition of such a term is a possibility for future development. Under most circumstances, the VAR residue model can perform just as well as a model that includes atmospheric terms. This is true in the following situations:

- Any historical filling scenario:
  Stations in the Delta respond very similarly to low-frequency events (lower than 1 cycle per day). For this reason, stage data at supporting stations give us almost all the low-frequency information required for Martinez. To illustrate the similarity between stations, Figure 8-8 shows the Martinez (RSAC054) and San Pablo (RSAC045) tidally-averaged stage during December 1995, one of the biggest wind events in recent history. The two stations rise and fall neatly in unison. The distance between the two curves (shown as a separate line in the plot) varies only 0.04 feet in each direction from a flat line, at the height of the event. Describing the effect of such an event at Martinez to this precision using a causal model between atmosphere and tide would be out of the question.

- Prediction made near the peak of an atmospheric event (point (c) on Figure 8-8):
  The VAR model captures and attenuates most events, as long as they are "in progress" at the time the prediction is made.

- Prediction made many days before the onset of an atmospheric event (point (a) on Figure 8-8):
  In this case, the atmospheric effects cannot be predicted at all. Thus, there is no way that atmospheric information can be put into the model successfully even if there were a term to accommodate it.

- Seasonal atmospheric characteristics (e.g. summer winds):
  The effects of the current season are manifest in the residual at the start of the prediction and are implicitly "in the model." Of course, this assumes that the VAR model is used to predict over an interval that is short compared to the length of the season.

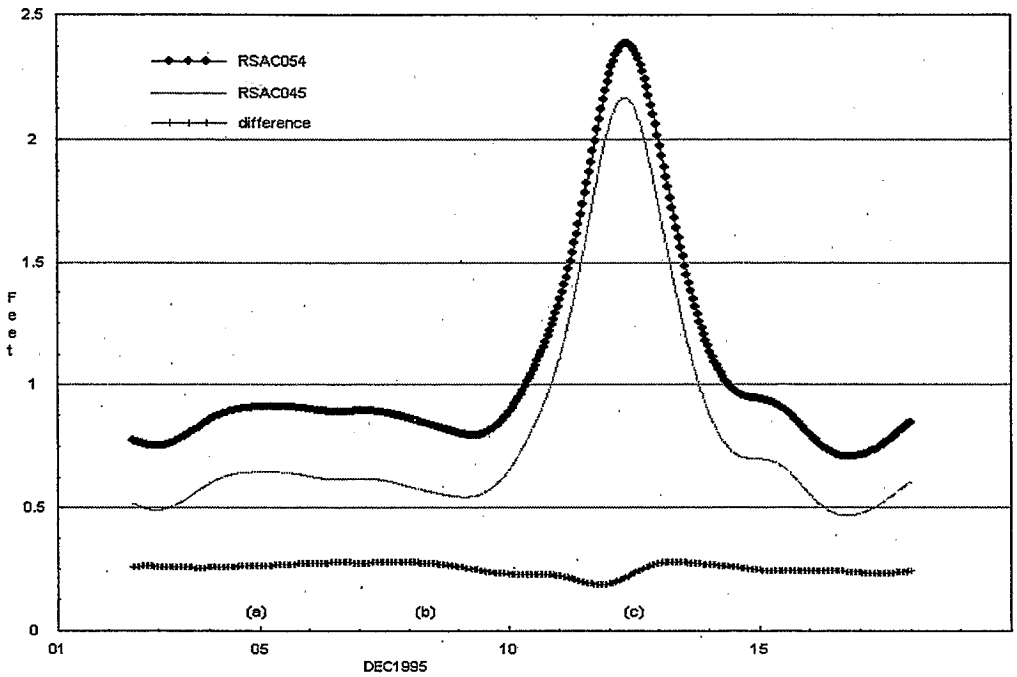

Figure 8-8: Tidally filtered stage during a major atmospherically-driven event.

Where improvement of the model is possible is when a prediction is made several days before an important, forecastable atmospheric event, as in point (b) on Figure 8-8 (recall that this was the case in the earlier example shown in Figure 8-5). The author is investigating the relationship between stage changes and barometric forecasts to see if the inclusion of a pressure term is warranted.

8.5.2 14-Day Cycling

A nonlinear interaction between solar and lunar constituents excites oscillations with a period of just over 14 days (this interaction is not the same as the "spring-neap" cycle, which is a linear interaction, but tends to be synchronized with this cycle). The oscillations are largely responsible for the "filling and draining" of the Delta which has been observed to be important in salinity transport. Oscillations of this periodicity are handled in equilibrium tide theory, standard NOS and DWR models, and in the model presented above inasmuch as they are captured in long-period constituent Mf.

Figure 8-9 shows low-passed stage over six months for RSAC054. Two-week cycling is apparent, but a sinusoidal representation seems unsatisfactory. In fact the Mf harmonic term fit to the full record (on stations where it was deemed significant) has an amplitude of about 0.1 foot, which means that it is not picking up variations of the magnitude that are obvious from inspection of Figure 8-9.

An alternate way of incorporating the 14-day term is to absorb it as a cyclical autoregressive part of the model. From Figure 8-9, we can see that characteristics tend to

repeat between adjacent two week cycles and this would be represented using standard VAR or ARMA terms – a common practice in seasonal hydrologic models (Bras, 1993). Figure 8-9 suggests that the improvement in medium-term prediction accuracy (forecasts of 14-20 days) might be as much as a half a foot. The downside to including 14-day periodicity is that the model will become much larger and more cumbersome.

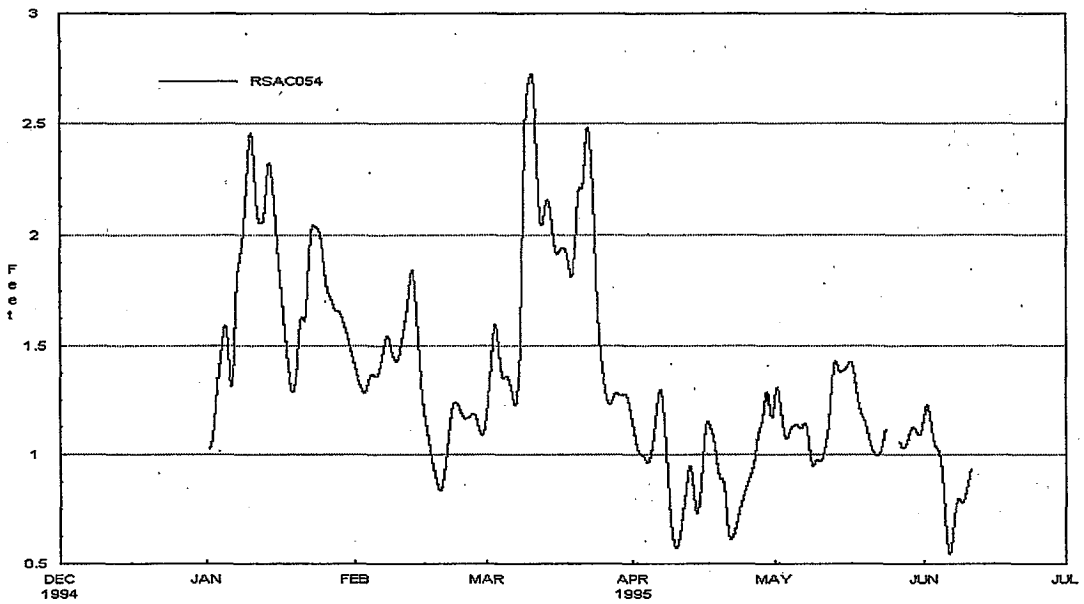

Figure 8-9: Tidally averaged stage at Martinez over six months.

8.6 Conclusions

Adding a tide residue model to augment ordinary astronomical models increases the quality of prediction and smoothing. The VAR model suggested here produces fill-in (smoothed) values that are extremely accurate, regardless of events which take place during the fill-in period. The model also improves prediction of tides, with the largest improvements during the first week. Both types of estimates tend to have close to zero error when compared to the observed data, so that the transition between real and estimated data is smooth.

This section has also examined two potential shortcomings of the VAR approach. The first is that the VAR does not employ a causal representation of atmospheric forcing on tides. This affects accuracy if the prediction is made just before the onset of a foreseeable storm. In almost all other situations involving astronomical events, it has been shown why explicit consideration of such forcing cannot benefit the model.

The second shortcoming of the VAR approach is that it uses a naïve representation of 14-day period variations. These oscillations are not really sinusoidal and attempts to model them as such usually fail to capture their full amplitude. A possible remedy for

this shortcoming is to include a cyclical but slowly varying 14-day term in the VAR model. This will improve medium forecasts.

8.7 References

Bras, R.L. and I. Rodriguez-Iturbe. (1993). *Random Functions and Hydrology*. Dover.

Godin, Gabriel. (1972). *The Analysis of Tides*. University of Toronto Press.

Hamilton, J. D. (1994). *Time Series Analysis*. University Press.

Harvey, A.C. (1994). *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge University Press.

Huynh, H.T. (1993). "Accurate Monotone Cubic Interpolation." *SIAM J. Numer. Analysis*. **30**(1), pp 57-100.

McLachlan, G.J. and Thriyambakam, K. (1997). *The EM Algorithm and Extensions*. Wiley.

Mardia, K.V., J.T. Kent, and J.M. Bibby. (1979). *Multivariate analysis*. Academic Press.

Munk, W.H. and D.E. Cartwright. (1966). "Tidal Spectroscopy and Prediction." *Phil. Trans. Roy. Soc.* (London), Ser. A **259**,533-81.

Schureman, P. (1941). *Manual of Harmonic Analysis and Prediction of Tides, Special Publication.* No. 98, U.S. Coast and Geodetic Survey.

Zetler, B.D. (1982). *Computer Applications to Tides in the National Ocean Survey, Supplement to Manual of Harmonic Analysis and Prediction of Tides (Special Publication No. 98)*. National Ocean Survey (NOAA).

9 Dissolved Oxygen Modeling Using DSM2-QUAL

9.1 Introduction

DSM2 is capable of simulating the dynamics of primary production including dissolved oxygen, phytoplankton, nutrients, and temperature. A single water quality variable or any combination of 11 water quality variables can be modeled as specified by the user. Changes in mass of constituents because of decay, growth, and biochemical transformations are simulated using interconstituent relationships derived from the literature (see references). This year the Delta Modeling Section has prepared to calibrate DSM2 for dissolved oxygen, focusing on the San Joaquin River near Stockton. Dissolved oxygen levels frequently fall below 5 mg/l, especially during the warm months. There is concern that low DO levels may adversely affect resident fish and other aquatic life. Low dissolved oxygen levels can cause physiological stress to fish and can block upstream migration of salmon.

DSM2-QUAL was updated to reflect changes in hydrodynamics and general input/output modules. This involved a few changes in the computer code and several test simulations. Code changes include an increase in array size representing constituent variables and keeping model output units and the observed data units consistent. QUAL's capability to use non-zero initial conditions ("warm start") for 11 constituents and to generate DSS output of all the constituents was tested and verified after necessary changes in input/output. This work is in progress.

9.2 Data Requirements

Simulation of dissolved oxygen requires information on water temperature, BOD, chlorophyll, organic nitrogen, ammonia nitrogen, nitrite nitrogen, nitrate nitrogen, organic phosphorus, dissolved phosphorus (ortho-phosphate), and EC in the Delta. Continuous sources of data were available for DO, temperature, and EC at hourly intervals for some stations near model boundaries. These data provide boundary information needed by DSM2. At most stations in the Delta, only grab samples are available, usually on a biweekly or monthly interval, which will be used as initial conditions for the model. Since continuous data were not available at Vernalis (RSAN112), hourly averaged values of DO, EC, and temperature available from the nearby station at Mossdale (RSAN087) will be used to approximate these quantities for the boundary inflow at Vernalis. Data on effluent flows from the City of Stockton's Regional Wastewater Control Facility (RWCF) were obtained from the Stockton

Municipal Utilities District. An estimate of water qualities of agricultural drainage returns at internal Delta locations was prepared. Simulation of water temperature requires hourly values of air temperature, wetbulb temperature, wind speed, cloud cover, and atmospheric pressure. Climate data representing the above were purchased from the National Climatic Data Center, and are being converted to DSS format required by DSM2.

9.3 Comparison of DO at Mossdale & Stockton

While much of the time observed DO at Stockton (RSAN058) seems to fall to very low levels, during dry periods or when Mossdale DO goes down, there were times when this trend did not happen. Based upon examination of observed water and air temperature at these locations, flows at Vernalis, and stage at Martinez, it appears that the influence of air temperature and tide tend to be more dominant when Vernalis flows are low. This may have caused some dips and rises in the DO pattern for a few months in 1994 and 1997. The barrier at head of Old River installed in spring 1997 seems to have improved dissolved oxygen levels at the Stockton station, i.e. approach the DO levels at Mossdale. This seemed true also for spring 1994, but because some data were missing, this could not be verified for the whole of the period when the barrier was operating. DO plots for 1994 and 1997 are presented in Figures 9-1 and 9-2.

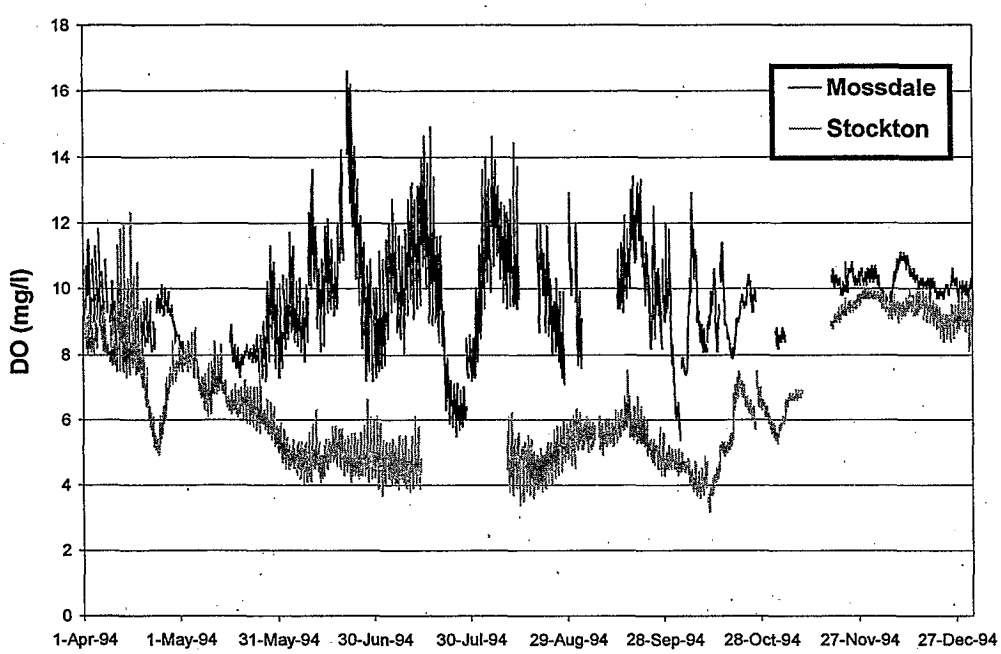

Figure 9-1: Dissolved Oxygen in the San Joaquin River (1994).

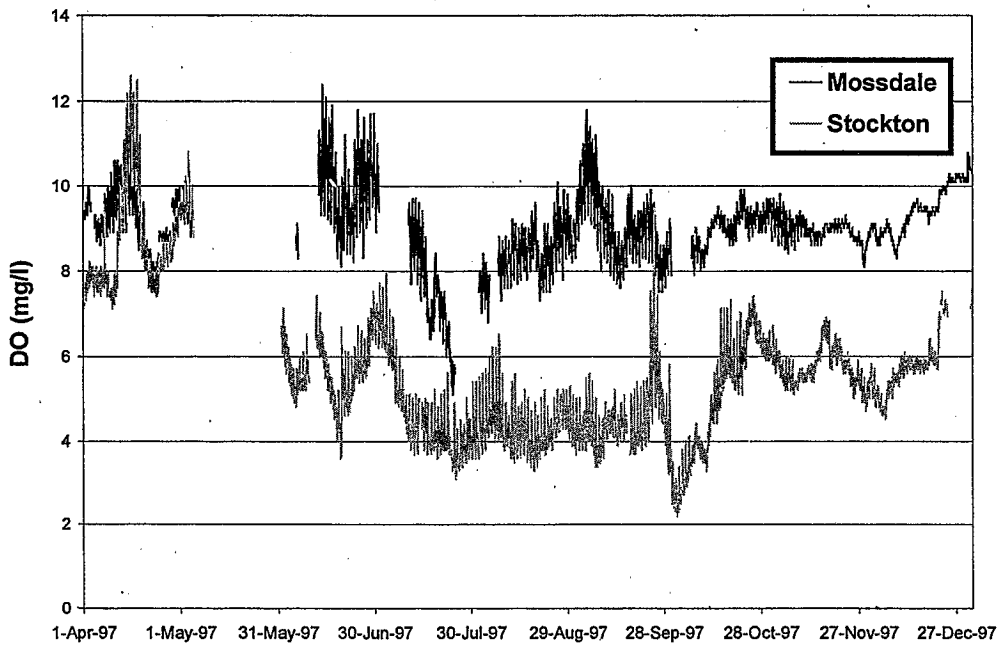

Figure 9-2: Dissolved Oxygen in the San Joaquin River (1997).

9.4 Calibration and Validation of DO

Based upon the examination of observed hourly temperature and DO for recent years at Mossdale, Stockton and Martinez, calibration/validation of DO will focus on the period from October 1997 through November 1998. The summer-fall period from 1998 is shown in Figure 9-3. Most other periods have blocks of data continuously missing both at Stockton and Mossdale at the same time, or air and water temperature missing at Mossdale at the same time.

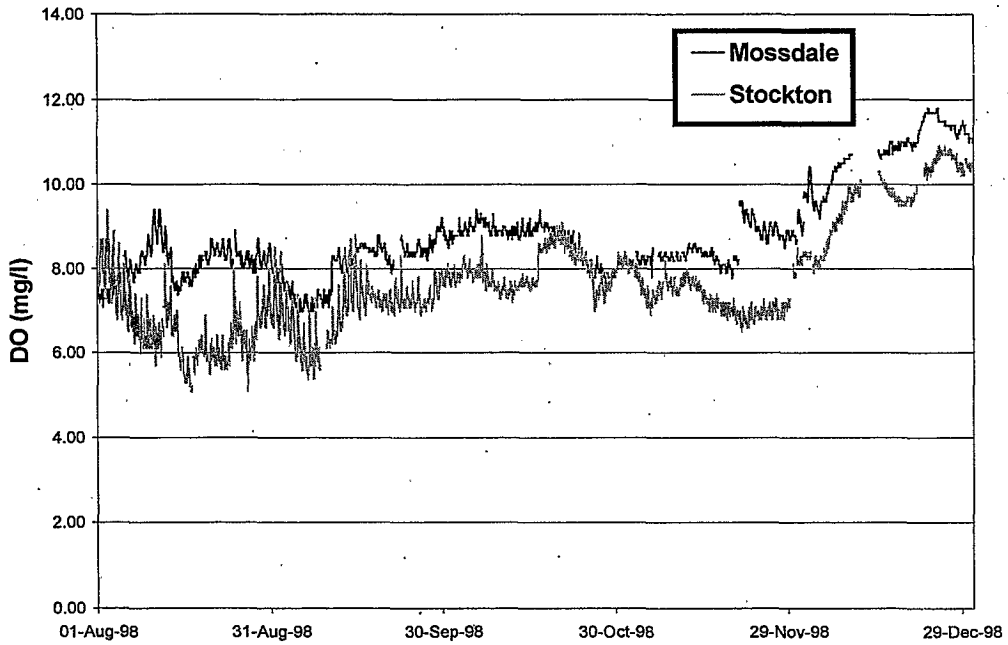

Figure 9-3: Dissolved Oxygen in the San Joaquin River (1998).

As a start, simulations were made with simplified hydrologic and water quality boundary conditions. Tests showed that the impact of increasing air temperature on dissolved oxygen varied significantly with location. The impact was higher in the interior Delta than at channels closer to model boundaries.

DSM2 calibration of EC is being conducted using the revised Delta grid that incorporates the latest bathymetry of the Delta (see Chapter 10). Calibration of dissolved oxygen will be based on that grid. Test simulations based on the revised grid showed that a few changes are necessary in the current QUAL input set-up. In addition to the data described above, physical, chemical and biological rate coefficients describing reaction kinetics are required as model input. Some of these coefficients are kept constant throughout the system; some are varied by location. Most of these coefficients are temperature dependent. For calibration of dissolved oxygen in the San Joaquin River, priorities will be put on the following parameters as control knobs: algae growth rate, algae settling rate, reaeration rate, sediment oxygen demand rate, light extinction coefficient, decay rates for ammonia nitrogen, organic nitrogen and BOD. Nitrification rate and the rates at which oxygen is produced from photosynthesis, and is lost to respiration, will also be used as knobs. The effect of the mixing coefficient on DO levels will be examined. Depending upon the results of the above calibration process, evaporation coefficient and dust attenuation factor that affect water temperature will also be examined. Sensitivity to initial and boundary quality conditions, nutrient levels of

agricultural drainage and those from the City of Stockton's RWCF effluent will be evaluated.

9.5 Future Directions

The Delta Modeling Section plans to coordinate its attempts to develop tools, in this case a calibrated model, with the San Joaquin River Dissolved Oxygen Total Maximum Daily Load (TMDL) Stakeholder process. The model should help identify the main factors that contribute to low dissolved oxygen situation in this reach of the San Joaquin River. Through evaluations of different scenarios, the model can aid in developing potential management strategies to address water quality degradation. The TMDL process was started because the State is required by federal law to establish limits on discharges that adversely affect dissolved oxygen in the San Joaquin River from all sources (e.g. cities, agriculture, industry, etc.). The TMDL Steering Committee consists of stakeholders from industry, agriculture, cities and state government agencies and has been meeting monthly since January 1999. Details on the meetings and the status of the TMDL process are available from `http://www.sjrtmdl.org/`.

9.6 References

Bowie, G.L., W.B. Mills, D.B. Porcella, C.L. Campbell, J.R. Pagenkopt, G.L. Rupp, K.M. Johnson, P.W.H. Chan, and S.A. Gherini. (1985). *Rates, Constants and Kinetics Formulations in Surface Water Quality Modeling, 2nd Ed.* US EPA. Athens, Georgia. EPA 600/3-85/040.

Brown, L.C. and T.O. Barnwell. (1987). *The Enhanced Stream Water Quality Models QUAL2E and QUAL2E-UNCAS; Documentation and Users Manual.* US EPA. Athens, Georgia. EPA 600/3-87/007.

O'Conner, D.J. and W.E. Dobbins. (1956). "Mechanism of Reaeration in Natural Streams." *Journal of Sanitary Enginering Div.*, ASCE. 82(6), 1-30.

Orlob, G.T. and N. Marjanovic. (1989). *Heat Exchange. Chapter 5 in Mathematical Submodels in Water Quality Systems.* ed. S.E. Jorgensen and M.J. Gromiec, Elsevier Pub.

Rajbhandari, H.L. (1995). Dynamic Simulation of Water Quality in Surface Water Systems Utilizing a Lagrangian Reference Frame. Ph.D. Dissertation, University of California, Davis. Davis, California.

10 DSM2 Calibration

10.1 Introduction

The Delta Modeling Section has been working with IEP (Interagency Ecological Program) with the goal of calibrating DSM2-Hydro and DSM2-Qual. Several agencies are contributing to this process. These include DWR (Delta Modeling Section, ESO, and O&M), USGS, USBR, Contra Costa Water District, Metropolitan Water District, and Stanford University. The first DSM2 calibration was performed in 1997 by Delta Modeling Section staff. This calibrated version has been used since fall 1997. Since then a significant amount of geometry and flow data have become available. It became evident that a new round of calibration would be justifiable. The DSM2 IEP-Project Work Team (PWT) was formed in late 1998. The group laid out the tasks needed for a successful calibration. The results for first iteration of DSM2-HYDRO became available in summer of 1999. At this point calibration is still in process. IEP-PWT hopes to have HYDRO and QUAL calibrated by the middle of summer 2000. The following is a brief discussion of the various aspects of this calibration process.

10.2 Calibration

IEP-PWT decided early on to make the DSM2 calibration an open process. The group laid down the strategy for having the results of all of the calibration runs available on the IEP Internet web-site. The staff from DWR's ESO developed a set of routines to make this process fully automated. The reader is referred to the following web-site for detailed information regarding the calibration, including the model results for all the iteration cycles done up to date:

```
http://www.iep.water.ca.gov/dsm2pwt/
```

At this Web site, a person can click on a map showing locations in the Delta where field and model data are available. A plot includes the results of the latest iteration, the results from the particular iteration cycle which the team believes had the best fit, along with the field data. The team looked for incremental improvements in matching the field data.

10.2.1 HYDRO

Four different time periods were selected for calibration of DSM2-HYDRO:

1. May 1988,
2. April 1997,
3. April 1998, and
4. September 1998.

This decision was based on the availability of the flow and stage data throughout the Delta.

Two sets of plots were generated: 1) instantaneous and 2) tidal-average. This enabled the team to check the tidal amplitudes, phase, mean flows, and the flow splits. In order to easily quantify how well the model matches the field data, some error indexes were defined and their values were computed and shown on all the plots. For specific definition of these error indexes see:

http://www.iep.water.ca.gov/dsm2pwt/calibrate/error_index.GIF

Any calibration effort requires manipulation of some model parameters. These are usually parameters which are not directly measurable in the field. For HYDRO the calibration parameter was chosen to be the Manning's n coefficient, which describes the friction characteristics of the channels. In HYDRO every channel may be assigned a unique Manning's n coefficient. To reduce the number of degrees of freedom, the Delta was divided into a series of about fifty groups of channels, with all the channels in a group having a single value of Manning's n coefficient. For a map showing the geographical location of these groups see:

http://www.iep.water.ca.gov/dsm2pwt/calibrate/chnl_groupsI.html

An idea that was suggested and later implemented was to change the network grid representing the Delta. The majority of the changes in the grid corresponds to the Western Delta, and mainly affected the way some of the large bodies of water are simulated. In the old grid, these open water areas were simulated as a reservoir, which basically acts like a tank. The disadvantage of the reservoir assumption is that any water parcels that enter are completely mixed in a single-time step. In the new grid, some of these open water areas are being simulated as a series of connecting wide channels. The new model grid map (as shown in Figure 10-1) is available as a 'zipped' version and can be downloaded from the IEP Web site at:

http://www.iep.water.ca.gov/dsm2pwt/calibrate/

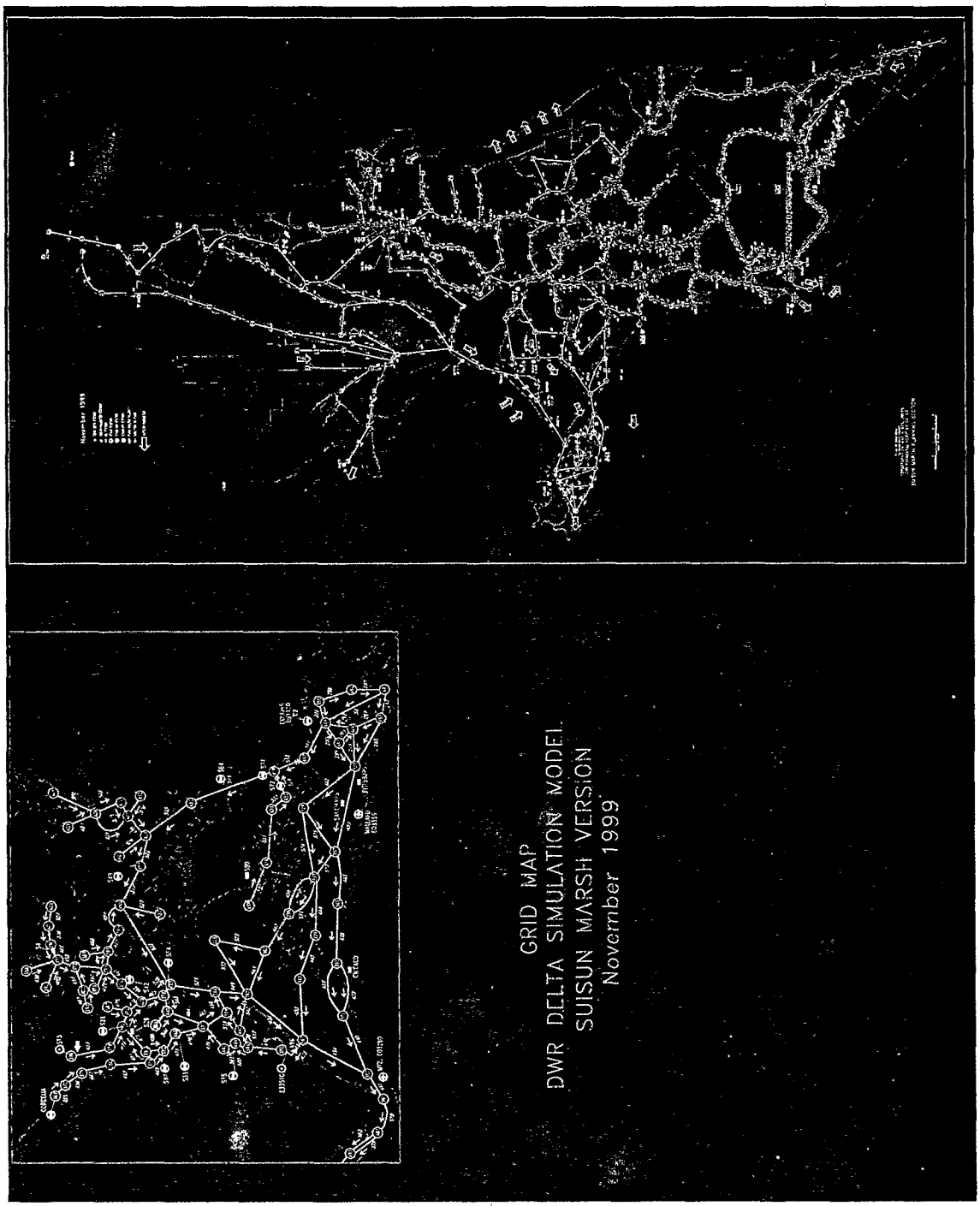

Figure 10-1: New DSM2 Grid (as of November 1999).

As of June 2000, there have been 49 calibration iteration cycles. The IEP-PWT members routinely attend conference calls to discuss the results of the latest runs and decide on the direction of any changes to the calibration parameters. The magnitudes of the incremental improvements in the model results are becoming smaller as expected.

Overall, the results of the latest run show major improvements in the model predictions compared to the previous calibrated version.

10.2.2 QUAL

The calibration of DSM2-QUAL has begun. The staff from the Delta Modeling Section is setting up the model runs and the interface needed to show the results. Unlike HYDRO, QUAL will need more than a few days to warm up. In fact it usually takes two to six months to wash out the impact of the initial water quality assumption. IEP-PWT decided to calibrate QUAL for the three consecutive years from 1992 to 1994. The most interesting periods for QUAL calibration are when noticeable salinity intrusion from the ocean occur. 1992-1994 is considered a dry period and contains three to four times where salinity rises and falls.

The calibration parameter for QUAL is the dispersion factor. The dispersion factor accounts for the process of mixing salinity between two neighboring parcels of water. Two runs have already been completed using dispersion factors of 0.25 and 1.0 (constant for the whole Delta). The following web-site contains a clickable map allowing the user to plot model results and the field data:

`http://wwwdelmod.water.ca.gov/studies/calibration/`

Once the process of calibration is under way, results from the QUAL runs may dictate the need for further calibration of HYDRO. QUAL is sensitive to small changes in flow especially during dry periods (low net Delta outflow). At this point, it is expected that the process of calibration for both models can take a few more months. By fall 2000, it is expected that the Delta Modeling Section will switch to using the new calibrated version of DSM2.

11 DSM2-QUAL Initialization

11.1 Introduction

Formally, physically-based models such as DSM2-HYDRO and DSM2-QUAL require initial and boundary conditions. These models march through time, using boundary conditions and physical conservation laws to update the state of the delta from one time step to the next. The marching proceeds from an initial state, whose influence is gradually lost as the simulation proceeds and the system "loses memory" of its initial condition.

In practical modeling, there are circumstances when initial conditions can be neglected (i.e., specified arbitrarily). When the period to be modeled is long compared to system memory, the initial condition influences only a small fraction of the run. In such cases, the model is often "cold started" from a numerically convenient initial condition. For example, HYDRO is almost "cold started," because it loses memory of a previous state within a day or two – a period much shorter than that of most modeling applications.

Water quality models such as QUAL have a longer memory of several months, which is not short in context of real-time applications of several weeks. At the end of a 14-day salinity simulation, much of the salt in the interior delta will have originated from somewhere within the model domain. The contribution of the initial salinity field must be accounted for, but this is a technical challenge – because at any moment salinity can only be known, with error, at a few dozen monitoring stations.

This report begins by introducing the Lagrangian mechanics of QUAL and discussing some of the interplay between various sources of error in the model. The subsequent sections survey options available for quality model initialization and introduce an optimization-based method that increases short-medium term accuracy. Finally, three methods are tested, and the results attest to the importance of the initial condition and the inadequacy of methods that do not incorporate field data.

11.2 Transport in QUAL

Figure 11-1 is a rudimentary portrayal of transport mechanics as modeled in QUAL. QUAL is written from a Lagrangian viewpoint with a moving frame of reference – the volume in a channel is subdivided into parcels, and these parcels are tracked as they move together with the local flow velocity. This movement as a "train" of parcels is known as "advection." When water enters the system, a new parcel is created to

accommodate the inflow and its concentration is taken from the boundary condition. In Figure 11-1, if the water is moving toward the right, Parcel 0 would represent an inflow from a boundary on the left side. When a parcel fully exits the system it is destroyed (Parcel 5 is a candidate if the right side is a boundary). Dispersion and other "mixing" processes are depicted in QUAL as a volume exchange of fluid between neighboring parcels at each time step. Finally, an observer as shown in Figure 11-1 as being fixed and "watching the train cars go by."

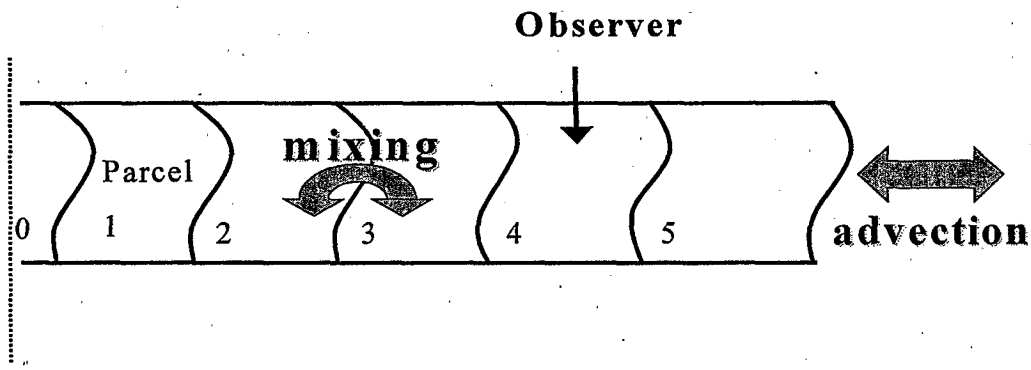

Figure 11-1: Advection and mixing of parcels in a channel.

Neither advection nor mixing is a speedy mechanism for carrying salt inland. Parcels flowing in from the ocean boundary tend to exit during the subsequent ebb or later in the spring-neap cycle. Ordinarily, for salt to persist in the delta, parcels making an incursion into the delta must mix with the water in the interior before leaving. In the simplistic example of Figure 11-1, Parcel 0 might appear during a (rightward) inflow, mix a bit with Parcel 1 and then leave again. Eventually, some of the new salt will disperse toward the right. In the meantime, the observer sees the salt from the interior parcels (perhaps Parcels 3-4) as they oscillate back and forth with the tide.

11.3 Sources of Error

Based on the discussion so far, quality simulation error can be attributed to three sources: initial conditions, boundary conditions, and model error.

11.3.1 Initial Conditions

At any time, the state of the Delta can be estimated by a snapshot of the 20-30 water quality monitoring sites. Estimates of initial conditions will be at gage accuracy near monitoring stations, worse in between. "Gage accuracy" here is meant to encompass instrument accuracy, representativeness of the cross-section, and the deficiencies of using EC as a surrogate for salinity.

The resolution of the initial condition in QUAL is limited. Although it is convenient to talk about initializing parcels (parcels being the fundamental unit of the model), the scale of the initial condition in QUAL is really the channel. One value may be assigned per channel, and is installed in all its parcels – they soon become different as they mix from water from other channels. To the extent that this resolution is adequate, it is because:

1. There are hundreds of channels;
2. The monitoring network is too sparse to justify higher resolution; and
3. Salinity is gradually varying.

The relative importance of initial condition error decreases over time.

11.3.2 Boundary Conditions

Water quality is monitored regularly at all the important boundaries. During historical periods, boundary error depends on "gage accuracy" (as defined above). In the "future" portion of a real-time run, flow and salinity boundary conditions are predicted with ever-decreasing accuracy. Thus, the magnitude of boundary error increases over time unless the run is based purely on historical data.

11.3.3 Model Error

The model itself introduces error. Even if boundary and initial conditions are specified perfectly, the one-dimensional approximation, uncertainty over parameter estimates and other numerical imperfections will distort the solution. This happens slowly.

"Model error" must be distinguished from "model operating on boundary error or on initial error." As an example of the latter, imagine that the initial salinity has been underestimated and we use the model to simulate the opening of the Delta Cross Channel (such a situation will arise later in the test problem). In such an experiment, the cross-channel opening may cause a milder change because the model mistakenly thinks that salinity is low to begin with. Such a result has a downside and an upside. The downside is that the model underestimates the effect of the delta cross channel operation. The upside is that two wrongs make a right: salinity started out too low and by not decreasing as much will end up closer to the correct value. Such negative feedback is common and as a result, model error tends to be bound in the long run. "Long-term model error" is a concept that is nebulous and problem-dependent, but is nevertheless useful. Long-term model error depends on the quality of calibration (an IEP project is underway to improve long term accuracy, see Chapter 10 of this report).

11.4 Initialization Strategies

The preliminary ideas above can now be applied to survey initialization strategies. Three additional terms will be used to describe the timing of the run (see Figure 11-2):

- t_s will be used to specify the model time at the beginning of the simulation,
- t_0 will be used to specify "now" or the beginning of the prediction phase of the simulation, and
- "spin-up" will refer to the period between t_s and t_0 in which the model warms up (if such a period is required).

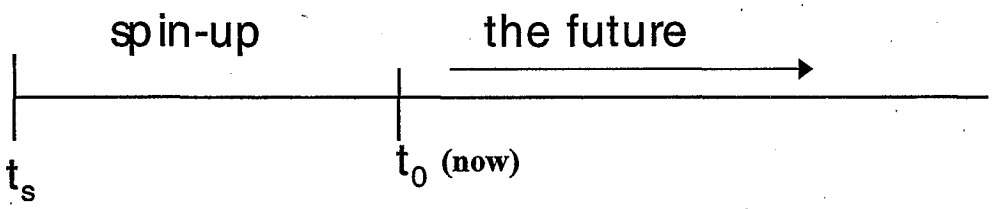

Figure 11-2: Timing of a real time run with spin-up.

11.4.1 Cold Start with Long Spin-Up

A common strategy for real-time applications is to start the model from an arbitrary initial condition and let it spin up for a long period (four-six months of model time is typical). The long initialization period eliminates initial condition error at t_0, but in exchange for a long period of cumulative model error and boundary error. In the illustration in Figure 11-3, the cold start begins with very large error, because of the arbitrary initial condition. Gradually, the initial error dissipates and the model approaches long-term model error. This is the best that the memory-loss model can do, and there is no recourse at t_0 if the model doesn't match recent observations.

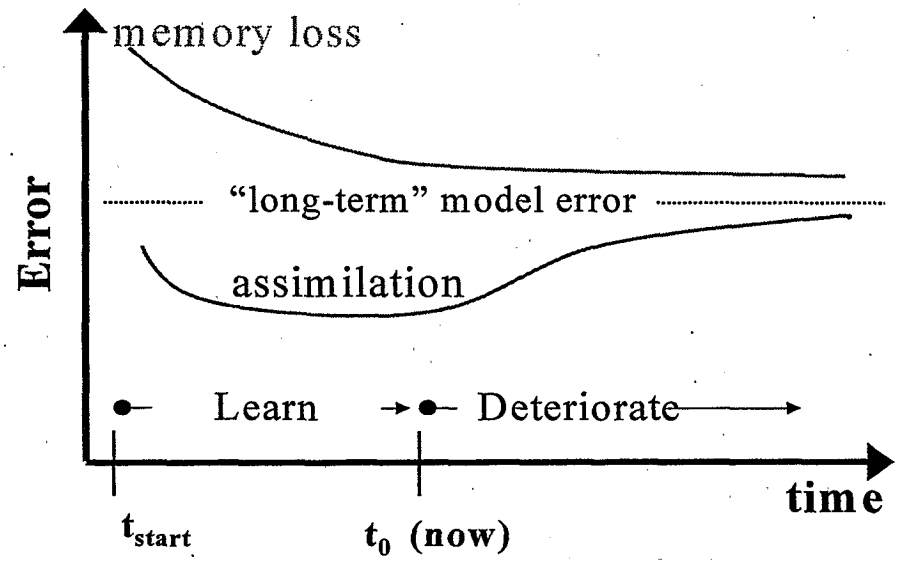

Figure 11-3: Error evolution under "memory loss" and "data assimilation" strategies.

The cold start is attractive because it is simple and skirts the technical difficulties of specifying a network-wide initial condition with limited data. Its biggest drawback is that it is less accurate than other strategies. It can not do better than long term model accuracy. One conclusion of this report is that the cold start is not sufficiently accurate for most real-time applications.

The time-efficiency of the cold start depends on whether the run is performed in isolation or as part of a regular program. A six-month water quality initialization requires not only a six-month water quality run, but also a six-month supporting hydrodynamic simulation to establish a flow field. The hydrodynamic run is a speed bottleneck – hydrodynamic runs of this length currently take a long time to perform (several hours using a 300 MHz machine). If, however, the quality simulation is part of a continuous program of real-time modeling – say once a week – then previous hydrodynamic results can be borrowed and only a modest incremental hydrodynamic run is needed. In this case, the cold start is fast.

11.4.2 Snapshot Initialization

Snapshot initialization is conceptually simple, and corresponds to the intuitive interpretation of initial conditions. The modeler takes the 20-30 or so observations that are available in the Delta at any one instant and interpolates their values over the full network. In snapshot initialization, there is no formal spin-up period, although it is wise to allow a few hours to smooth transient effects at start up. All snapshot methods are vulnerable to station noise and reliability problems because they are based on one value per station.

The snapshot approach is complicated only by the many interpolation schemes that are possible. Here is a brief summary:

11.4.2.1 Patch Schemes

A zero-degree approximation assigns the value at a monitoring station to every channel within a prescribed "area of influence." Members of the Delta Modeling Section have developed such a scheme.

Figure 11-4 shows the areas of influence ascribed to each monitoring station for a patch schematic based on 31 stations. Note that each patch must contain a monitoring station. The patch-based method is sensitive to the monitoring network size and configuration as well as the assignment of areas of influence – and must be reapportioned when there are missing data. Figure 11-5 shows results for a patch scheme based on 21 stations versus results for 28 stations (from the 31-patch map shown in Figure 11-4, but with 3 stations not available). For the first three weeks, the results are different. Patch-based snapshots of fewer than 20 stations are thought to be inadequate.

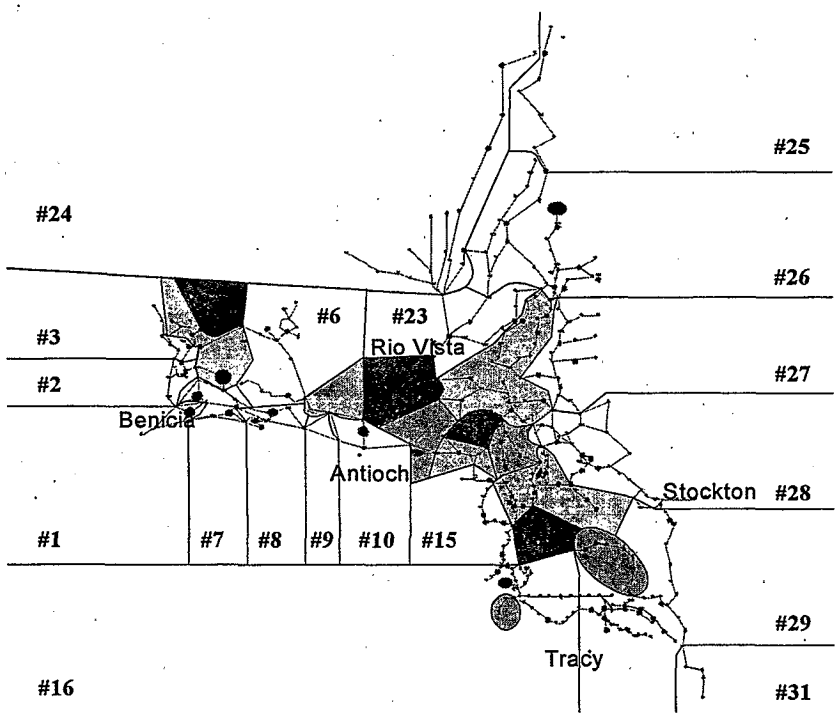

Figure 11-4: Regions of constant initial EC used in the patch-based snapshot scheme.

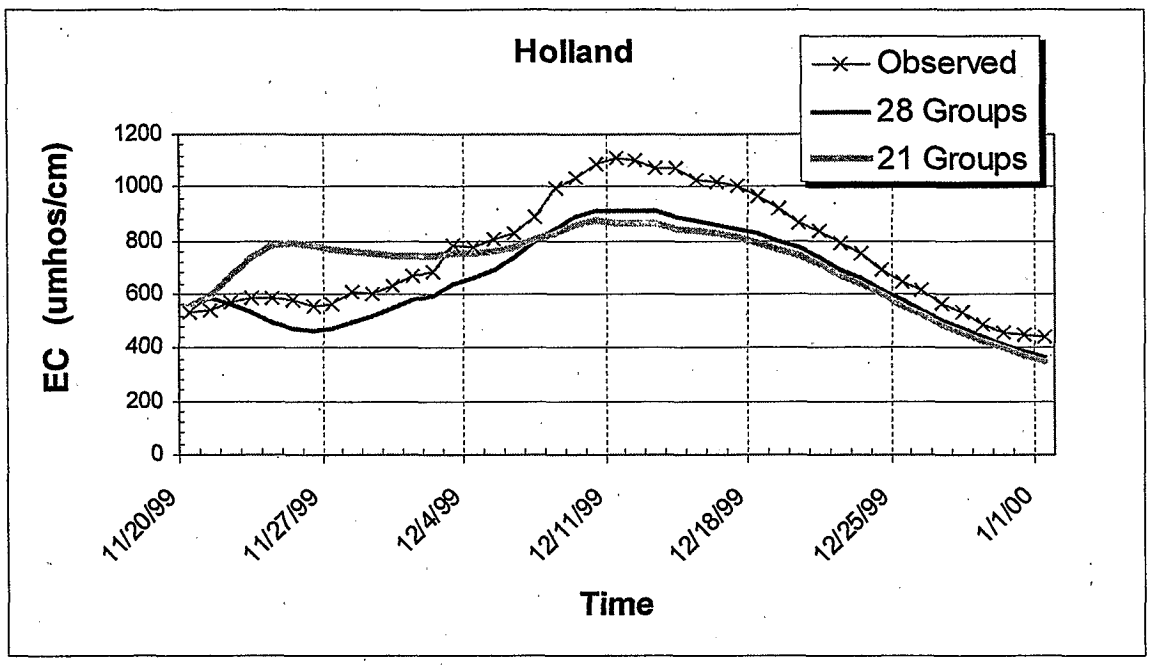

Figure 11-5: EC results and observed data for simulations based on 28 groups and 21 groups.

Because they are based on conditions at one instant, the patch-based snapshot scheme is slightly sensitive to the choice of start time – the snapshot seems to work better at some points in the tidal cycle than others, though this point has not been investigated in detail. This sensitivity to start time was found to be more acute when the patch-based scheme is applied using tidally-averaged EC instead of instantaneous values; for this reason, the use of tidal averages to start the model was abandoned.

11.4.2.2 Higher Order Splines

A smoother initial salinity field between stations can be achieved using higher order approximations (linear interpolation, splines, etc). One difficulty with higher order methods is how to specify them over a network. One proposal by this author is to use a high order monotonic spline to connect stations along the big river systems (Sacramento, San Joaquin, and Old River) that have multiple observations. Linear interpolation would be used in the cross-channels that bridge these main rivers, and zero-order estimates in dead-end sloughs and attached water bodies. Such a scheme cannot be theoretically justified without a sufficiently dense monitoring network, assumptions on the shape of the salinity profile and a high signal-to-noise ratio. The assumption of high signal-noise is violated in the South Delta – particularly in the upper San Joaquin where ambient salinity is low but the observation stations pick up spikes of salinity due to agricultural return flows. Higher order schemes were not tested for this paper, but if the noise problem is surmounted they seem likely to compensate for some of the deficiencies of zero-order interpolants.

11.4.2.3 Physical "Smoothing"

As an alternative to filling the salinity field using interpolation, some modelers insert observations into the model using artificial dispersion/diffusion. The advection component of the quality model is turned off, and data are allowed to diffuse in from the observation stations over time, creating a smooth field. No such methods were tested for this paper.

11.4.3 Optimized Initial Condition

So far the snapshot schemes (and the patch schemes in particular) have two main drawbacks. First, the wrong value may be installed between stations if we choose the wrong interpolation method or if stations are too far apart. Second, the schemes are vulnerable to station noise because they use one value per station.

Data assimilation schemes seek to get around these problems by selecting the initial condition that gives the "best" fit to all monitoring data observed over a spin-up period of several days. The number of data used greatly increases: most delta monitoring stations report hourly values, so that the number of observations available in a three day spin-up is about:

3 days* 24 observations per day per station* 30 stations = 2,160 observations.

The use of sequential measurements helps compensate for the spatial sparseness of the monitoring network. For an illustration see Figure 11-6. Assume that the parcels are initialized at time t using data from Stations 1 and 2 for Parcels 4 and 1 respectively, and that some interpolation scheme is used in between (assume that parcels can be initialized individually). At time (t+1) all the parcels have moved to the right and some mixing has occurred. Parcel 3 now resides at the monitoring station. If a new observation is available at Station 1 at time (t+1), a discrepancy will be observed between model and data, and we will learn something about what should have been put in Parcel 3 in the first place. If we continue to fit data over several tide cycles and observe many parcels, we may improve our initial estimates for all the parcels in the channel. The improvement will be largest for parcels whose tidal excursion brings them under the monitoring station soon after the start of the run; however, since these parcels mix with the others (and are influenced by them), we can indirectly learn something about all of them. So far, no mention has been made of the optimization algorithm or implementation details. These are discussed in detail in section 11.5.

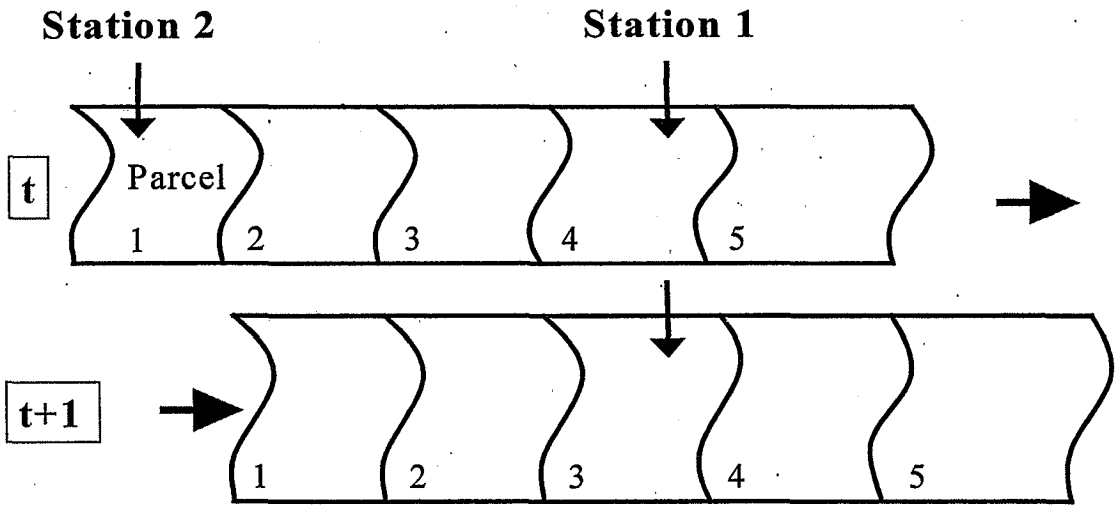

Figure 11-6: Exploiting sequential measurements to gather spatial information.

The evolution of error in a data assimilation scheme is pictured conceptually in Figure 11-3– the scheme is characterized by a period of learning leading up to t_0 followed by a period of deterioration as the model slowly approaches "long-term model error." Note that Figure 11-3 contains an abuse of scale: the spin-up period (the distance between t_s and t_0) is about 6 months for the memory-loss scheme and about 3 days for assimilation schemes.

The optimization-based method is much faster than a "cold start" in isolation, but slower than a "cold start" that is part of a regular real-time program (see the discussion of "cold start" efficiency above). The optimization-based method is slower than snapshot methods, although it saves on some of the decisions and experimentation required in timing the "snapshot." The optimization component is global and well behaved.

11.4.4 Recursive data assimilation

One more way to take advantage of new observations is to assimilate data using a recursive filter (Kalman filter). In the example from Figure 11-6, we would correct the parcels on the spot at time (t+1) and send them on their way, instead of going back and trying to find a better initial condition. The Kalman filter does this, nudging the model towards the data recursively. The Kalman filter is based on the premise that both model and data contain error, and the relative magnitude determines the extent of the "nudge."

Recursive filtering trivializes the initial condition – the idea is not to find a good initial condition at all, but rather to run the simulation continuously, gradually folding in new data. Thus, a recursive filter may be the approach most deserving of the label "real-time tool." Unfortunately, there are several disadvantages:

- The filter is invasive to the model. The Kalman filter requires computation inside QUAL (derivatives of all the computational steps). This is particularly difficult because of the use of sub-time steps.

- Standard Kalman filters are not appropriate for a Lagrangian problem. QUAL has a variable number of model states (depending on how many parcels are created and destroyed at each time step). The recently developed Kalman filters for rank-deficient "descriptor" systems can be used to address this problem, but the filters must be coded from scratch.

- The Kalman filter is a recursive computational tool, not a complete model. The approach requires an error model for both observations and QUAL, which would have to be described and estimated from scratch.

Despite the disadvantages, the Kalman filter (perhaps in concert with other startup methods) may eventually prove to be a valuable tool.

11.5 Initial Condition Optimization

11.5.1 Introduction and Formulation

A conceptual picture of initial condition optimization through data assimilation was given in previous sections. The optimization of initial conditions is not a new idea – it can be thought of as a parameter estimation problem and treated using numerical optimal control or adjoint data assimilation algorithms. In light of some of the specifics of QUAL, these methods are not as efficient as the technique advocated here based on the principle of superposition (and related to transfer functions).

Given a flow field, transport is governed by a set of partial differential equations (PDE) which are linear. This means that the principle of superposition applies – see Fisher (1972) or any book on linear systems. The superposition principle says that the sum of any two solutions of the PDE (or scalar multiples of these solutions) will also satisfy the PDE. It is a property of the underlying equations, not a model representation – although it is well preserved by QUAL.

Superposition is illustrated in Figure 11-7. Assume that we have a channel with an initial condition in two patches, one with a concentration of 3 "units," and the other with a concentration of 2 (top of figure). A known boundary condition B(t) is imposed at both ends of the channel for all time. We observe the evolution of concentration at points throughout the channel. The concentration fields thus observed can be obtained from the following sub-problems:

- one problem with the same boundary conditions as the original problem, but an initial condition of zero everywhere;
- one problem with zero boundary conditions, a nominal concentration of 1 in the first patch and a concentration of zero in the second patch; and
- one problem with zero boundary conditions, a concentration of zero in the first patch and a nominal concentration of 1 in the second patch.

To obtain the solution to the original problem, we sum the first problem, 3.0 times the second problem and 2.0 times the third problem.

Now, turn the problem around, and assume that the initial concentrations of the two patches are not known. From the foregoing discussion, the three fundamental problems above can be combined linearly to construct the solution to any initial value problem involving the same two patches and boundary conditions. Let us seek the linear combination that gives the best fit (say, in a least-squares sense) to the observations, which are available in various places in the channel. Regardless of whether the true initial condition is really in two patches, this is the optimal "two patch" initial condition.

The optimization problem is as follows, written in terms of n patches instead of two:

$$\min_{\alpha_i} \sum_{x,t} w_x \left[C(x,t) - \hat{C}(x,t) \right]^2 \tag{0.6}$$

such that

$$\hat{C}(x,t) = \hat{C}_b(x,t) + \sum_{i=1}^{n} \alpha_i \hat{C}_i(x,t) \tag{0.7}$$

$$\alpha_i > 0 \quad i = 1..n \tag{0.8}$$

where:

- $C(x,t)$ and $\hat{C}(x,t)$ are respectively the observed and modeled salinity concentrations at discrete station locations (x) and time steps (t),
- $\hat{C}_b(x,t)$ is the salinity concentration field obtained from the sub-problem with boundary conditions and zero initial conditions,
- $\hat{C}_i(x,t)$ is the salinity concentration field obtained from the sub-problem with zero boundary conditions and a nominal initial concentration of unity in patch (i),
- w_i is a weight depending on station scaling (unity in the experiments presented in this paper),
- α_i is the initial condition in the *i*th patch, or, alternatively, the linear coefficient of the *i*th basic sub-problem. This is constrained to be positive because it represents a positive physical quantity.

Equations (0.6) through (0.8) comprise a convex linear-quadratic program, a type of problem for which there are reliable solution algorithms (routine QPROG from the IMSL library was used for the prototype). The problem has a single, global minimum. Time/station combinations with missing data can simply be omitted from the fit.

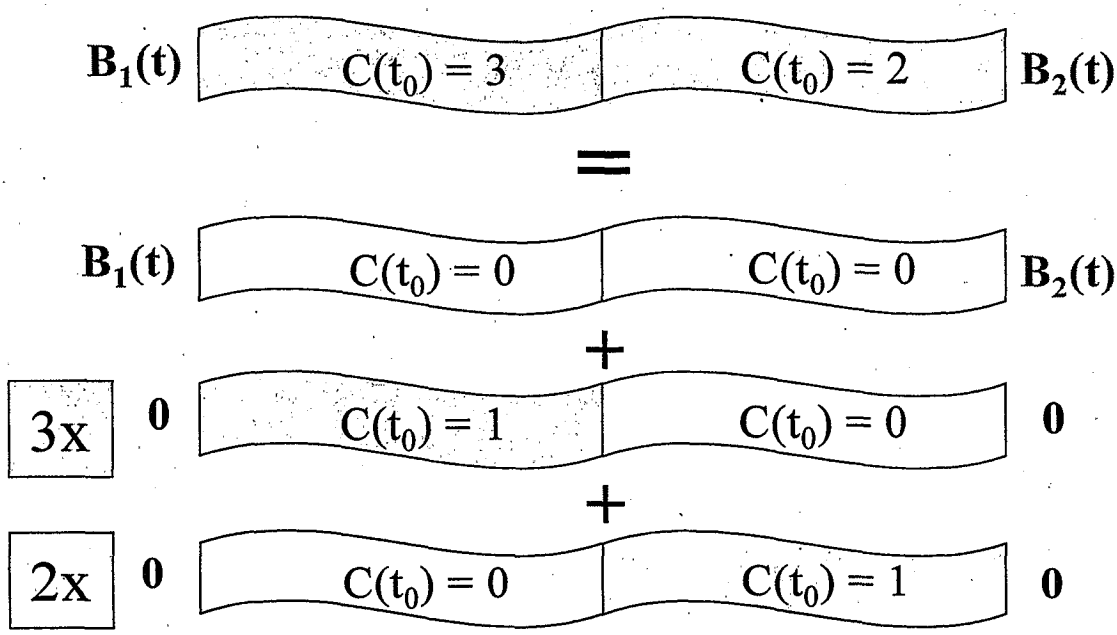

Figure 11-7: Example of superposition. The top initial value problem has been decomposed into a linear combination of simpler sub-problems.

11.5.2 Monotonicity Constraints

The main stems of the Sacramento and the San Joaquin Rivers tend to have salinity that decreases monotonically from the ocean boundary. One-dimensional transport dynamics will always yield monotonic concentration gradients (in the long run) when net flow is toward the ocean boundary and the ocean boundary is the contributor of salt. Such conditions exist along most of the Sacramento, and usually along the San Joaquin to about Jersey Point (upstream on the San Joaquin, the network of channels allows circulatory flow, and salinity spikes appear due to agricultural return flow). In the regions where monotonicity is appropriate, it may be enforced by inequality relationships between neighboring patches:

$$C_i < C_j \quad (i,j) \in \mathrm{M} \tag{0.9}$$

where the elements of set M are pairs of patches with a monotonic relationship.

11.5.3 Stability Constraints

The optimization method works best in regions that are well monitored or have a healthy advective exchange with a monitoring station. In small remote areas, the concentration may be less "knowable" and subject to erratic values under optimization. The remote regions may either be lumped together with neighboring patches that are better monitored or they may be bound to these "main" patches by not allowing more than a specified fraction of difference (they can also be fixed *a priori* – see Modifications and Extensions below). A stability constraint may be written:

$$\text{abs}(C_i - C_j) < kC_i \quad (i,j) \in \text{N} \tag{0.10}$$

where the elements of N are pairs of "main" (i) and "bound" (j) patches and k is the maximum fraction that the "bound" channel is allowed to deviate from the "main" channel. In practice, this expression is decomposed into two linear constraints.

In the original implementation of the optimization scheme, stability constraints were used to bind neighboring channels, reservoirs to nearby channels (reservoirs are treated just like channels in this scheme) and dead end sloughs to main channels, using $k = 0.2$. In regions that are adequately monitored, stability constraints should be left out so that reservoirs and sloughs can act in their natural role as buffers against salinity change.

11.5.4 Formulation Flexibility

Under the optimization method suggested here, the number of QUAL runs is fixed – once the basic sub-problems have been assembled, the constraints or weights of the optimization problem can be re-specified without additional model runs. The optimization problem itself takes only a fraction of a second to solve.

11.5.5 Efficient Computation and Choosing Patch Size

Under the optimization scheme, we are no longer required to define one patch per monitoring station. In fact, we are at liberty to define "patches" up to the finest resolution available for a QUAL initial condition: one patch per channel.

In practice, we would never want to do this, because it would require hundreds of "fundamental" solutions and the initial condition would be over-resolved compared to the monitoring network. In the schematic used for the tests in this report, there are 82 patches, including 68 channels and 14 reservoirs (which can be treated the same as patches). The areas near Martinez are highly resolved (the salinity gradient is steep and well monitored), but the patches in remote regions in the south delta are lumped in much bigger groups.

There are still 82 patches, and currently performing 82 basic QUAL runs would take a long time – even though the hydrodynamics portion is only run once and the QUAL runs

are conducted only over a 3-day spin up period. Fortunately, it is possible to use QUAL's multiple constituent capability to drastically reduce the number of runs required. "Pseudo-constituents" can be used to represent EC originating from different patches (this idea is also used in "source fingerprinting" studies done by the section), so that about ten patches can be simulated at once. The total computational burden is one three day HYDRO run and about 8-10 QUAL runs. This amount of computation is comparable to that of other methods, and on a 300 MHz machine the method takes about 20 minutes.

11.5.6 Modifications and Extensions

11.5.6.1 Basic Solutions

The methods discussed above use linear combinations of patches for initial conditions. However, we are at liberty to construct solutions from other fundamental building blocks. The only restriction (due to superposition) is that each "building block" can only be adjusted using a scalar multiple.

Instead of separating the basic sub-problems into "boundary" and "patch" constituents, we may use the more general idea of "fixed" and "optimized" regions. The initial condition does not have to be unknown in every region. In some cases, the modeler may want to stipulate salinity *a priori* over part of the model domain. A run using the "fixed" region then becomes analogous to what was previously the "boundary only" run.

Similarly, we can incorporate results from a previous model run to assist in regions that are poorly monitored. The output from a previous run would be part of the "fixed" part of the solution. This solution would be adjusted locally in an amount of detail proportional to the density of the monitoring network. This would mean adjusting each channel near the Western boundary, but adjusting crudely or not at all in remote dead-end sloughs.

11.5.6.2 Filtered Values

It is also possible to use filtered salinity for the optimization instead of instantaneous values. The motivation for using filtered values is to take advantage of the fact that QUAL is more accurate in a tidally-averaged sense than in an instantaneous sense (because of the averaging of phase-dependent errors). Since such averaging is just a type of linear filter, superposition still applies to the output and the optimization method can be applied directly. If averages are used, information loss must be prevented. This requires the use of an appropriate low-alias filtration of the salinity data whose output is defined at the same hourly time step as the original series (rather than some sort of daily average with only one output value defined per day).

11.5.6.3 Monitoring Network

Finally, the results from the optimization method provide a tool for assessing how good the water quality monitoring network is for purposes of model initialization. The optimization method works well in regions that are monitored directly and in regions that are not but have a high advective exchange with a location that is well monitored. By assimilating data over time, the technique "leverages" the information that is available – and its failures indicate locations that require better gage coverage.

11.6 Comparison

A test was conducted between three of the techniques discussed in this report: the cold start, the patch-based snapshot scheme, and the optimization method developed in this report. The test period was November 20 - December 10, 1999, a period of high salinity and dynamic operation, the discussion of which focused attention on real time modeling (it was the subject of the January 2000 Bay Delta Modeling Forum workshop). The Delta Cross Channel was closed on November 26, 1999.

The experiment was conducted with historical boundary flows, in order to control for the influence of boundary error. The cold start spin-up began on February 10, 1999. The snapshot of the "patches" was taken just after midnight on November 20. The optimization spin-up period was November 20-23 with the first four hours ignored to allow the development of a smooth salinity gradient (refining the length of the optimization period is an important item for further development).

Figure 11-8 and Figure 11-9 compare tidally-averaged EC results from the three initialization methods with field data at 6 monitoring stations. Going into the period of interest, the cold start uniformly underestimates salinity at all the stations, sometimes by as much as 50 percent. At some stations, the reaction to the Delta Cross Channel and other operations is accurate despite a persistent negative bias; for instance, at Jersey Point, the shape of the curve matches the shape of the observed curve even though it is way too low. At other stations, the effect of the Delta Cross Channel opening has been attenuated. This is the situation described above where a salinity-controlling operation doesn't do as much because salinity is already low at the outset. The snapshot and optimized methods dramatically reduce bias.

To compare the patch-snapshot method to the optimization-based method, we must neglect the spin-up period November 20-23. After this period, the optimization method yields a further 4-10 days of reduced error compared to the snapshot method at most stations. The reduction in error is most striking at Jersey Point and Rio Vista – but for different reasons. At Jersey Point, the setting is perfect for optimization. The surrounding region is monitored adequately, but the optimization method can help fill in unknown areas. In addition, Jersey Point is remote from the boundary – so the initial condition is critical.

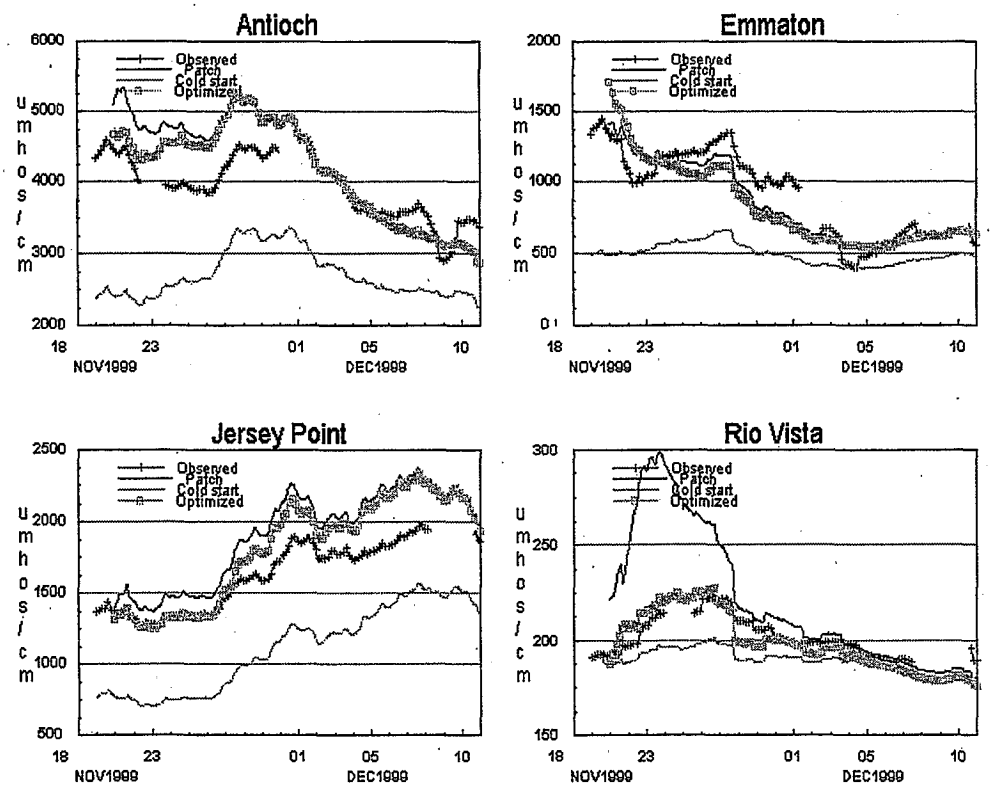

Figure 11-8: Tidally averaged EC for the three methods of model initialization.

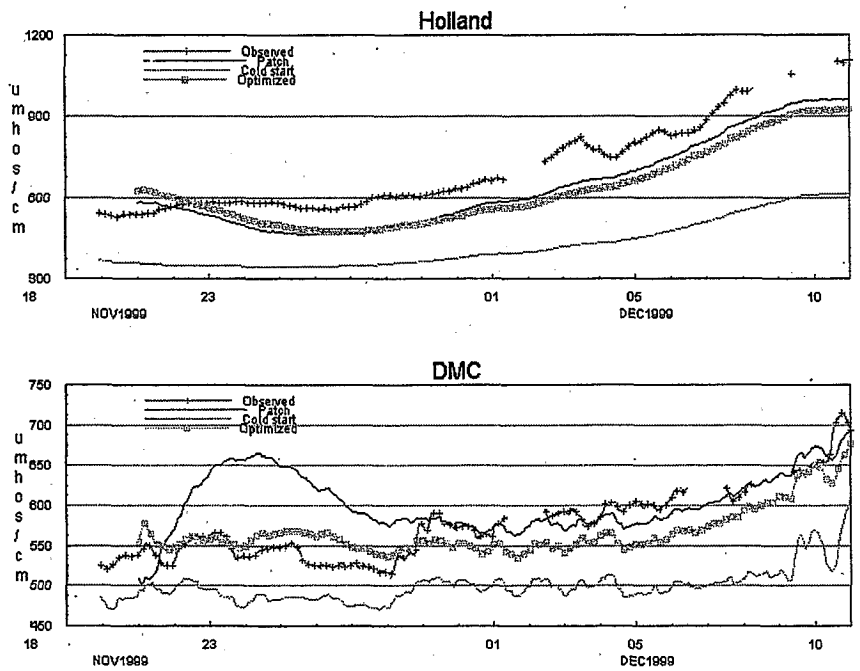

Figure 11-9: Comparison of tidally averaged EC at Holland Tract and DMC.

At Rio Vista (and also DMC), the difference between the two methods is because of a delayed period of overestimation using the patch-snapshot method. This is apparently a hazard of its coarse, zero-degree approximation. In the experiment, a downstream patch with high initial salinity advected towards the Rio Vista area with lower initial salinity, causing an abrupt increase (although the reader is encouraged to look at the scale of the Rio Vista plot). Had the initial model salinity varied more smoothly from high to low, the water that reached Rio Vista would have been less saline and the transition would have been gradual. The optimization-based method achieves smoothness through high resolution. Since in the optimization method the initial patches do not have to contain a monitoring station, many patches can be used (one or two channels per patch was used in this region).

The results at Emmaton are interesting because the patch-snapshot method performed better than the optimization method did, even during the November 20-23 optimization period. Emmaton was partially "sacrificed" in the optimization process in order to reduce error at other stations. This is the only location that the author is aware of where this happened. One reason why this might occur is that station magnitudes vary quite a bit in this region, and unweighted least squared error was used in the objective function. Unweighted least squares preferentially accommodates stations with large magnitudes of error, and Emmaton generally has an EC (and error) almost an order of magnitude lower than that of its western neighbors such as Antioch. The production version of the optimizer has since been rewritten to allow station weighting.

Finally, the salinity trajectories from the snapshot-patch scheme and the optimization-based scheme tend to converge by the end of the test period, particularly in the north (remoteness of the boundary affects convergence). Their mutual approach to the cold start solution, on the other hand, seems to have barely begun. Evidently, "independence of initial conditions" is a relative concept – it takes months for two runs with very distinct initial conditions to converge, but only 10-20 days for two runs with fairly similar initial conditions to do so. It seems safe to assume that runs based on any of the various data-based initialization strategies surveyed in this report (i.e., all of the techniques except the cold start) will be nearly indistinguishable after the third week of simulation.

11.7 Conclusions

This report has surveyed methods of initializing real-time model runs. The most important points are as follows:

- Error arises from a variety of sources. Modelers should be attentive to the "weak point" of the modeling process at various stages of a run. In real-time salinity modeling, initial condition error is an important "weak point," but one that can be ameliorated.
- It takes a fairly long time for DSM2 to reach its "long-term" level of error – much longer than the duration of a real-time run. When the model is initialized well,

memory works to our advantage and the model deteriorates to "long-term" levels slowly. When the model is initialized poorly, the initial condition will contribute to error for the whole run.

- The optimization-based method described in this article uses a stream of data over time at each station to arrive at a satisfactory initial condition. The scheme plays to QUAL's strengths, including preservation of superposition and multi-constituent capabilities.

- Alternative tools are available. The rudimentary patch-based snapshot scheme works well considering its simplicity. Recursive filtering is the most conceptually pleasing "real time" tool, but is cumbersome to implement in QUAL.

- The "cold start" (reliance on long-term memory loss) is not sufficiently accurate for real-time applications.

11.8 Reference

Fischer, H.B., E.J. List, R.C.Y. Koh, J. Imberger, and N.H. Brooks. (1972). *Mixing in Inland and Coastal Waters*. Academic Press, San Diego.

Printed by
Department of Water Resources
Reprographics